电脑

4
1987

中国软件行业协会会刊

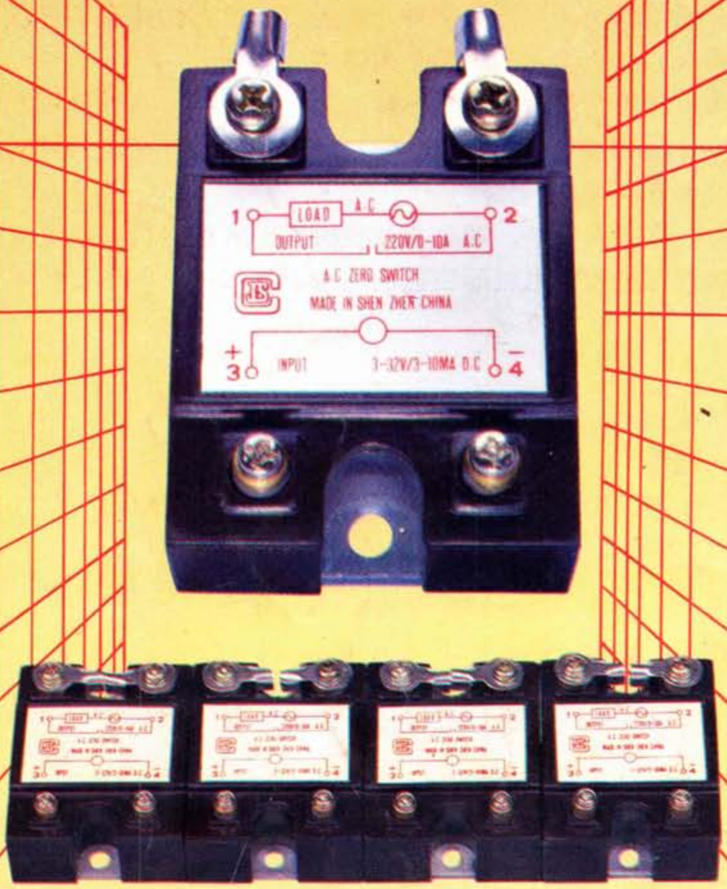


微机激光排版交叉编辑系统

COMPUTER

交流过零无触点开关

带光电隔离单相 220 V／ 10 A



京深电脑有限公司 JING SHEN COMPUTER LTD.

地 址：广东省深圳市笋岗路八挂岭工业区 615 幢 3 楼

电话：40006 、40007 电挂：7528

邮 购 消 息

邮购进口元件、质量保证、信誉第一。单价以人民币元为单位、除注明外每次购件邮费伍角。

型号	单价	型号	极性	vcbo	vceo	ICMA	PCMW	FTMHZ	外型	封装	单价
三端稳压管		2SA673		35	35	0.5	0.4	50	TO—92		0.6
7805	2.6	1015	硅	50	50	0.15	0.4	80			0.6
7806	2.6	940		150	150	1.5	25				3.2
7808	2.6	2SB546	P	200	150	2	20		TO—220	塑	3.2
7812	2.6	566	N	70	50	4	40				3.2
7815	2.6	596	P	80	80	4	30				3.2
7818	2.6	BF423		250		0.1	0.83				1
7824	2.6	422		250		0.1	0.83				1
7905	2.6	2SC458		30	30	0.1	0.2	230	TO—92		0.6
7906	2.6	1730		30	15	0.05	0.25	1100			1.2
7908	2.6	1815		60	50	0.15	0.4	80			0.6
7912	2.6	1669		150	150	1.5	25				3.2
7915	2.6	1507	硅	300	300	0.2	15	80	TO—220		3.2
7918	2.6	2073		150	150	1.5	25				3.2
7924	2.6	2456		300	300	0.1	10	50	TO—126		3.2
LM317SP	7	2577		120	80	6	60				8.5
LM317K	12	2581		200	140	10	100				8.5
集成块		3505	N	900	700	6	80				8.5
Z80ACTC	18	2SD1397		1500	800	3.5	50				8.5
Z80AP10	18	1398		1500	800	5	50	带			8.5
Z80CPT	18	1425		1500	600	2.5	80	阻	TO—3PB		8.5
Z80ACPU	18	1426		1500	600	3.5	80	尼			8.5
4N39	3.6	1427		1500	600	5	80			封	8.5
LF353N	2.5	1439		1500	1500S	3	50				8.5
HM6116LP—3	11	1453		1500	600	3	50				8.5
P8088A	18	401	P	200	150	2	20		TO—220		3.2
UPB8214C	15	478		200	150	2	30				3.2
M5L8212P	12	200		1500	1500S	2.5	15				12
P8253—5	14	348		1500	400	7	50			金	12
D8243HC	13	869		1500	600	3.5	50	阻尼	TO—3		8
P8257	16	2SC1942		1500	800	3	50			属	8
PB8251R	30	BU208S		1500	700	7.5	60				12
P8259A	12	326A		900	400	6	75				8
P8237A—5	28	406	N	400	200	7	60				5
M5L8039—11	12	407		330	150	7	60				3.2
P8255A—5	14	407D		330	150	7	60	阻	TO—220	塑	5
D8035	14	806		400	200	8	60	尼			5
D5101LC	11	508A		1500	700	15	125		TO—218		13
ADC0809	18	BD135		45	45	1.5	12.5		TO—126		2.5
2732K	20	137		60	60	1.5	12.5				2.5
2764K	20	241		55	45	3	40		TO—2'20		3
27128	20	242	硅	55	45	3	40			封	3
27256	28	136	P	45	45	1.5	12.5		TO—126		2.5
2516JDL	22	138	N	60	60	1.5	12.5				2.5
LM324N	2	MJ2955	P	100	60	15	115				6
339N	3	2N3055	硅	100	60	15	115			金	5.5
393N	2.5	2SC2027	NPN	1500	800	5	50		TO—3		12
723N	3.5	BUX48A		1000	450	15	175			属	28
74LS32	1.5	2SC2761		450	400	30	200				39.5

深圳西乡震华电子器材经销部

邮购部地址：深圳市西乡龙珠路48号　电话：89136　电挂：3624　开户：宝安县工商行新安办

门市部地址：深圳市解放路63号　电话：36353　帐号：34—067001

交流过零无触点开关

带光电隔离单相220V／10A

交流固体无触点开关是我公司吸收国外先进技术研制成功的一种新产品。它具有无触点，高度可靠，使用期长，应用灵活等一般触点开关无法比拟的优点，能广泛地应用于各种控制低压输入，高压输出的大功率电路中。

1. 输入端与输出端控制是通过光电转换，无电的联系，从而使输入端与输出端有良好的隔离,特别适用计算机的开关量输出口,作一次性驱动，减少外界对计算机的干扰。是弱电控制强电的理想器件。

2. 零电压开通，零电流关断，减少负载（感性、容性）的起动电流，防止电冲击。

3. 避免一般触点开关因大电流而引起的触点表面氧化，跳火及粘连的故障，提高了接触的可靠性。

输入特性

输入	电压范围	3～32VDC
输入	电流范围	2～30mA

绝缘（输入与外壳、输出与外壳、输入与输出）　＞5MΩ

输出特性

输出电压范围	15～220 VAC（有效值）
输出电流范围	0.05～10AAC
导通时间	≤10mS
关断时间	≤10mS
温度	＜80°C

使用时应注意

1. 分清高压端、低压端，不可将其接反。
2. 低压部分，应分清正负两极，并根据电压范围，限流使用。
3. 此开关必须将底部固定在金属板，以便散热。
4. 输入与输出无电的联系，不得共地。
5. 高压部分要外接熔断器和保险丝。

目前，此开关已应用在计算机开关量的输出，打印机的电源开关，控制电动机等方面，防止干扰，已取得明显的效果。在工业自动化和家庭电器智能化将得到更广泛的应用。

引进日本松下生产线

胜风牌空调器

空气去湿机

——窗式、分离式、柜式、负离子空调器，恒温恒湿机、空气去湿机、系列品种齐备，最适合电脑室使用。

胜风、胜风、胜似春风！

SHENGFENG
胜风牌
出口商标
高宾牌

窗式空调器——

- 1983年～1985年连续三年获省优质奖
- 曾获外贸部优质出口商品《荣誉证书》
- 多次选送国际博览会

产品三包、代办托运、资料函索即赠，欢迎订购整机或套、散件

本厂在北京、上海、南京、杭州、郑州、武汉、长沙、成都、重庆、沈阳、青岛、桂林、梧州、汕头、湛江、海口设有维修服务中心

空调机型号

	型号	制冷量（大卡 时）	功能
窗口式	KC—12型	1200	冷
	KC—16型	1600	冷
	KC—18型	1800	冷
	KC—18D型		冷、暖
	KC—30型	3000	冷
	KC—35型	3500	冷
	KC—35D型		冷、暖
立柜式	B1660Ⅱ—IC型	7000	冷、暖
分离式	KG—32（室内）	3200	冷
	KW—32（室外）		

空气去湿机型号

型号	吸湿量（公斤 小时）
XSH—05型	0.5
XSH—3型	3





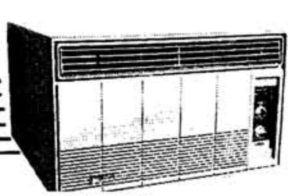


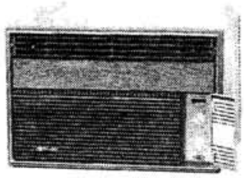
广州无线电专用设备厂

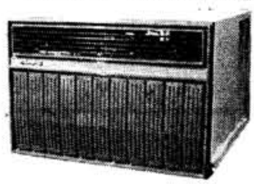
厂址：广州景泰坑　电话：总机663601、销售科662449

电挂：1743

展销部：广州海珠中路123号　电话：886870









电脑

（月　刊）

1987年

第四期（总第16期）

电脑应用

在ADCH——DOS的环境下实现IBM　PC双屏工作的一种方法……迟小龙（4）
一种菜单技术……毛　军（8）

软件纵横

CP/M工具软件ED程序剖析（四）……郭嵩山　曾超荣　郑吉丰（10）
怎样直接在CC　DOS汉字库中造字或修改汉字……李亦何（24）

有奖专栏

第七届海洋奖人选题目及作者……（17）

使用与维修

软磁盘驱动器维修经验一例……秦山秀　闵艺华（22）
IBM—PC/XT常见故障与维修（二）……张毅忠　郑存陆　罗家耀（27）

大学生之页

APPLE Ⅱ 高分辨率造型与色彩分析……陈中藩（34）

电脑辅助教学

电脑辅助抛体运动的教学……肖学璞　吴义琴（18）
绘Z＝F（x，y）立体图的一个简化程序……李兆丁（41）

中学天地

APPLE Ⅱ 低分辨率直方图显示及其打印技巧——显示和打印实用柱状统计图表的方法……黄伟民（14）
APPLE Ⅱ 高分辨图像的缩制……陈杰林（15）
一九八六年广东省青少年计算机程序设计竞赛部分试题的评述及参考解答……李冠英　邬家炜（38）

游戏乐园

迷宫问题新解……龙安川（31）

电脑用户

为充分利用打印纸而设计的程序……简志雄　傅雄俭（45）

万花筒

实现延长8位机喇叭发音时间的方法……张国治（26）
小鸟与松鼠……陈旭林（48）

厂家与产品

胜风牌空调器空气去湿机……（1）

交流过零无触点开关……………………………（封三）

简　讯

华穗分公司的改革值得推广
其成绩值得赞叹……………………邱达生（3）
CWY系列交流参数稳压器技术
交流订货会圆满结束………………郑潮文（9）
1987 IEEE亚洲电子学
会议即将在香港召开………………陈兴业（23）
抗干扰稳压器在广州问世…………………（48）
电脑运动鞋面世……………………何竞欧（45）

资　料

软件赠送及IC芯片供应…………………（21）
邮购消息…………………………（34）（封二）

启　事

敬告读者……………………………………（37）

责任编辑：　曾令宝　吴　军
美术编辑：　丘文东

主　　办　电子工业部计算机工业管理局
编　　辑　《电脑》编辑部
（地址：广州市石牌华南师范大学微电子学研究所）
出　　版　华南师范大学微电子学研究所
排　　版　电脑杂志社电脑激光排版中心
印　　刷　粤北印刷厂
总发行处　韶关市邮电局
订 阅 处　全国各地邮电局、所
刊　　号　46－115
定　　价　0.50元
出版日期　87年4月1日
广东省期刊登记证第118号
广告经营许可证穗工商广字004号

华穗分公司的改革值得推广

其成绩值得赞叹

全国计算机培训教材审编委员
广州智能工程研究会名誉会长　邱达生

中国计算机系统公司华穗分公司（华南计算机软件公司），在我国尤其在广州地区在推广应用计算机，开发计算机应用系统工程方面是成绩卓著的科研单位。

除了在计算机软、硬件的人材方面具有老中青的层次结构之外，取得这样成绩的重要原因是该公司领导人三年来坚定不移的改革，而且和所有试行改革的单位一样，经历了种种磨难，遇到种种阻力，但始终不渝地走在改革的道路上。

华穗分公司三年来开发了七十余个中小型的计算机系统工程及软件包。有五个工程被选参加了全国计算机应用展览会，有一个工程获得全国计算机应用二等奖，即将进行部级鉴定的工程数个及进行局级鉴定的工程数个。由于华穗分公司所进行的计算机工程的开发，直接促成的计算机销售额已近千万元。

华穗分公司受计算机用户的委托已在许多领域作出了自己的贡献：

1．在计算机与计算机之间的近距离和远距离通讯方面：

已实现了IBM43系列（中型机）与索拉机（小型机）的通讯；

已实现了索拉机与微机之间的通讯；

已实现了PDP－11系列机与微机之间的通讯；

已实现了VAX－11系列机与微机之间的通讯。

该公司以上述各通讯网络为基础，已研制出各类拓扑型的星形网及其应用系统。

2．它在工业自动控制方面：

承担了国家下达的三项重点工程（冶金、火电、水电计算机控制系统）。

已为鞍钢、武钢研制出板坯连铸的索拉机控制系统；（下转48页）

在ADCH—DOS的环境下实现IBM—PC双屏工作的一种方法

中国人民解放军
54410 部队技术股
迟小龙

所谓IBM—PC的双屏工作，是指在单个PC主机的条件下，同时使用单色监视器和彩色监视器显示信息资料的一种系统配置方式。大家知道，绿色屏幕的单色监视器具有最小的视觉疲劳以及较好的中西文字符分辨率等优点，但是，假如我们用它来显示彩色图形资料，效果自然是不如彩色监视器了。因此，如果用户的事务管理系统既有文字资料同时也有图形资料需要处理的话，那么无论采用何种单一的监视器作为文字和图形资料的显示输出，其效果都不是最佳的。所以，假如您想使系统能够兼顾单色监视器和彩色监视器各自的优点，使它能够在微机事务管理上做到图文并茂，采用双屏工作便是一种可行的办法。一般说来，在PC—DOS或CC—DOS的环境下实现双屏工作并不困难，使用PC—DOS提供的mode命令就可以实现。但是，若要在能够支持PC国标汉字显示卡的ADCH—DOS的环境下实现双屏工作，情况就要复杂一些了，因为ADCH—DOS不能完全支持mode命令以及诸如高级BASIC语言中的作图命令，因而使我们在监视器切换与图形程序的运行等方面受到限制，因此，若想在ADCH—DOS的环境下实现双屏工作，必须解决好上述两个方面的问题。本文根据我们在ADCH—DOS环境下实现IBM—PC双屏工作的实践体会，介绍一种在ADCH—DOS环境下实现双屏工作转换方法，供拥有ADCH—DOS进行双屏工作的用户参考。

一. ADCH—DOS介绍及其问题

在我们正式讨论在ADCH—DOS的环境下实现双屏工作的方法之前，有必要首先介绍一下ADCH—DOS及其存在的问题，以使我们对该系统能有一初步的了解，便于帮助我们认识在ADCH—DOS的环境下实现双屏工作的必要性。

ADCH—DOS系统是新近推出的一种能够在PC上支持国标汉字显示卡的先进中西文操作系统，它可以在分辨为720×350的单色监视器上每屏显示多达26×40个汉字字符，比目前流行的CC—DOS中文操作系统每屏11×40个汉字字符，字符容量提高了一倍半。因此，使用PC国标汉字显示卡和ADCH—DOS可以使用户在一屏上看到尽可能多的中文信息资料，同时由于汉字显示卡采用字符方式处理汉字，使汉字的显示速度大大提高，除此之外，ADCH—DOS还具有文件兼容性好等方面的优点。然而，尽管采用国标汉字显示卡和ADCH—DOS能使我们在微机事务管理方面获得许多良好的效益，但同时也给我们实现双屏工作带来一定的困难，这就是由于ADCH—DOS的设计者出于设计上的某种考虑，使得ADCH—DOS不能完全支持PC—DOS2.00以上版本中的mode命令以及高级BASIC语言中的全部作图命令，这是因为ADCH—DOS对ROM—BIOS中的显示驱动模块做了较大修改，对ROM—BIOS中的INT 10H类中断功能(以下简称10类中断功能)做了重新定义的缘故(虽然ADCH—DOS配有HBAS

IC文件，可以在单色监视器上支持720×350和640×200两种分辨率的单色图形显示，然而相信对于大多数人来说，彩色图形更有吸引力）。这也就是为什么说在ADCH—DOS环境下实现双屏工作比在PC—DOS或CC—DOS环境下实现双屏工作要复杂一些的原因所在。为此，我们对ADCH—DOS的显示模块进行了粗略分析，并找到了一种能够实现监视器切换和兼容高级BASIC语言中全部作图命令的方法，采用这种方法后，我们就可以根据事务管理系统的需要，在ADCH—DOS的环境下得到图文并茂的效果了。

二．在ADCH—DOS环境下实现双屏工作的一种方法

通过上面对ADCH—DOS及其存在问题的简单介绍，我们可以看出，采用国标汉字显示卡和ADCH—DOS后，实现双屏工作的困难在于：①PC—DOS原有的监视器切换命令得不到完全支持。②高级BASIC语言中的作图命令得不到支持，因而，即使监视器切换成功也是枉然。显然，欲在ADCH—DOS的环境下实现双屏工作，关键是解决好这两个问题。下面，我们就分别谈谈这两个问题的解决办法并给出相应的源程序。

1．修改ADCH—DOS的显示驱动模块，解决系统对高级BASIC作图命令的支持

和CC—DOS一样，ADCH—DOS为了能够处理汉字信息，同样也是对ROM—BIOS中的显示驱动模块进行了修改，其中也包括了对10类中断功能的修改，我们知道，ROM—BIOS中的10类中断主要是用于选择各种显示驱动模式以及显示字符和决定光标位置，但ADCH—DOS由于设计上的某种考虑，对10类中断内用于选择各种显示驱动模式的功能（AH=00H，AL=0～7）做了限制，使限制后的10类中断不再具有40×25彩色与黑白字符方式、80×25彩色与黑白字符方式、320×200彩色与黑白图形方式、640×200黑白图形方式等6种显示驱动模式的功能选择，由于PC—DOS的mode选择监视器命令和高级BASIC语言中的作图命令都要涉及到10类中断内上述各种驱动模式的选择，故ADCH—DOS不能完全支持mode命令和高级BASIC语言中的作图命令就容易理解了。ADCH—DOS对ROM—BIOS中的10类中断功能的限制是通过对10类中断入口地址进行重新定义来实现的，新的10类中断入口地址是RAM中的0579：0DA1，当程序中出现10类中断调用时，系统将按照新设定的10类中断入口地址（0579：0DA1）调用做了修改的10类中断，而不是象通常那样按照ROM—BIOS中的10类中断入口地址（F000：F065）调用10类中断（在ADCH—DOS中，该地址为79类中断入口地址）。了解了上述情况后，我们就可以通过编制一些小程序来重新定义10类中断入口地址，使显示驱动模块再次从ADCH—DOS的RAM—BIOS上转到ROM—BIOS上，从而可使高级BASIC语言中的作图命令得到执行。下面是一段用于将显示驱动控制从ADCH—DOS的RAM—BIOS上转换到ROM—BIOS上的汇编源程序（用8088汇编语言编写，以下同）。

```
源程序                     注释
mov  aX, f000H   ;⎫ 置ROM-BIOS中的
mov  dS, aX      ;⎬ 10类中断入口地址。
mov  dX, f065H   ;⎭
mov  ah, 25H     ;⎫ 置中断功能号及中断向量
mov  al, 10H     ;⎭ 定义10类中断，中断入
int  21H         ;  口地址为dS：dX
mov  aX, 3       ;⎫ 起始80×25彩色字符
int  10H         ;⎭ 模式
mov  dX, 0       ;⎫ 置光标显示位置参数，显
mov  ah, 2       ;⎬ 示光标于0行0列。
int  10H         ;⎭
```

如果我们希望在图形资料显示完毕后，显示控制重新回到ADCH—DOS的RAM—BIOS上，可以通过运行下面这段程序来实现。

```
源程序                     注释
mov  aX, 0579H   ;⎫ 置10类中断在RAM-
mov  dS, aX      ;⎬ BIOS中的入口地址，
mov  DX, 0da1H   ;⎭ dS：dX为入口地址。
mov  ah, 25H     ;⎫ 置中断功能号及中断向量
mov  al, 10H     ;⎭ 定义10类中断。中断入
int  21H         ;  口地址为dS：dX。
mov  aX, 2       ;⎫ 起始80×25单色字符
int  79H         ;⎭ 显示驱动模式。
mov  ah, 0       ;⎫ 起始单色国标汉字显示卡
int  10H         ;⎭ 40×26汉字字符显示
                    模式显示光标于0行0列。

mov  dX, 0       ;⎫
mov  ah, 2       ;⎬
int  10H         ;⎭
```

注意·在返回ADCH—DOS的RAM—BI

OS 的过程中，除了把1 0 类中断入口地址重新定义到RAM—BI OS 中的1 0 类中断入口地址外，还必须通过7 9 类中断将显示驱动模式转为单色字符的显示驱动模式（假定转换前是在彩色监视器上使用彩色字符显示驱动模式工作），以使ROM—BI OS 中的单色字符显示驱动模式接管整个显示控制（这一点在双屏工作方式中很重要，因为不同的字符显示驱动模式将决定资料在屏幕上的滚动特性），接着起始ADC H—DOS 中a h =0 的1 0 类中断，用以使单色字符的显示驱动模式从8 0 ×2 5 的纯西文显示模式转换为4 0 ×2 6 的中文显示模式，a h =2 的1 0 类中断，用以决定光标在4 0 ×2 6 中文显示模式下在屏幕上的位置，其中d h 为列坐标，d l 为行坐标。

2，显示屏幕（监视器）的切换

应该说，当我们把1 0 类中断入口地址重新定义到ROM—BI OS 上后，高级B A S I C语言中的作图命令就可以被执行了，但是注意到ADC H—DOS 多使用单色监视器来显示汉字信息（主机板组合开关上的显示方式位设为单色方式），如果此时用户对显示屏幕不加以切换的话，那么原来是彩色的图形就只能具有一种单一的颜色（绿色）了。因此，如果用户的P C 机配有彩色图形监视器／适配器的话，那么最好是将彩色图形资料送往彩色监视器上显示，以取得较好的图形显示效果。但是，我们怎样才能在系统已设定使用单色监视器的情况下，将彩色图形资料送往彩色监视器上显示呢？让我们先看看计算机内部是怎样确定使用那个监视器工作的。在P C 机的主机板上有个8 位的微型组合开关（S WI T C H1 ），其上第5，6 位用来设定系统所使用的监视器类型，具体规定及其对应的逻辑值如表一所示：

表一：主机板微型开关与监视器类型选择

监视器类型	主机板微型开关状态	对应的逻辑值
无监视器	1 2 3 4 5 6 7 8 ON	0 0
彩色图形监视器（40×25）	1 2 3 4 5 6 7 8 ON	0 1
彩色图形监视器（80×25）	1 2 3 4 5 6 7 8 ON	1 0
单色监视器（80×25）	1 2 3 4 5 6 7 8 ON	1 1

当系统加电自举后，系统将根据开关的设定位置选择一个监视器作为系统的控制台监视器使用，但实际上，系统则是根据主机板微型开关在DOS 数据区中的一个对应字节（地址：0 0 0 0 ：0 4 1 0 ）中的逻辑值决定这一选择的。显然，如果我们能够修改0 0 0 0 ：0 4 1 0 字节中对应于监视器选择的比特位之逻辑值，我们就可以迫使系统改变正在使用的监视器的类型（此时主机板微型开关仍为单色监视器设定）。下面是一段将单色监视器显示转为彩色监视器显示的程序例子。

```
源程序                         注释
mov  ax, 0            ;  设定主机板微型开关
                         在DOS数据区
mov  es, ax           ;  中的地址es: di
mov  di, 0410H        ;  =0000: 041
mov  al, es: [di]     ;  0取es: di地址
                         单元之值
and  al, cfh          ;  修改监视器选择位之
or   20h              ;  逻辑值并送回DOS
mov  es: [di], al     ;  数据区。
mov  ax, 3            ;  起始80×25彩色
int  10h              ;  字符显示驱动模式。
```

三．一个在ADC H—D O S 环境下实现双屏显示的例子

本例子由DOS 批处理程序d o u b s c r e ，b a t （见程序1 ）、ADC H—DOS 单色监视器工作转P C —DOS 彩色监视器工作程序g t c o l o．c x c （见程序2 ）、P C —DOS 彩色监视器工作转ADC H—DOS 单色监视器工作程序g t m o n o．c x c （见程序3 ）以及使用高级B A S I C 语言编写的彩色图形程序g r a p h．b a s （只有文件名，也可以是用户自已的图形程序）等程序组成，并在DOS 命令状态下运行d o u b s c r e．b a t 批处理程序，双屏显示就可以实现了。

（1） d o u b s c r e · b a t 批处理程序

程序1

```
源程序                          注释
1. echo  off           ;  关屏幕回应。
2. gtcolo              ;  转彩色监视器工作。
3. basica  graph       ;  运行彩色图形程序。
4. gtmono              ;  转单色监视器工作。
5. echo  on            ;  开屏幕回应。
```

（2） 转P C —DOS 彩色监视器工作程序

程序2

程序2・gtcolo・exe程序

```
code    segment para public 'code'
        org 100h
main    proc far
        assume cs:code,es:code,ds:code
        jmp start
pap     db  0dh,0ah,"* * OK ! * *"
        db  0dh,0ah,"1986.6¥"
stack   segment para stack 'stack'
        db 64   dup(0)
stack   ends
start:  push ds
        mov ax,0
        push ax
        mov ax,cs
        mov es,ax
        mov ds,ax
        push es
        xor ax,ax
        mov es,ax
        mov ax,es:[0410H]
        and al,0cfh
        or  al,20h
        mov es:[0410H],ax
        pop es
        push ds
        mov ax,0f000h
        mov ds,ax
        mov dx,0f065h
        mov ax,2510h
        int 21h
        mov ah,0
        mov al,3
        int 10h
        pop ds
        mov dx,offset pap
        mov ah,09h
        int 21h
        mov dh,3
        mov dl,160
        mov ah,2
        int 10h
        ret
main    endp
code    ends
        end main
```

（3） 转ADCH—DOS单色监视器工作程序

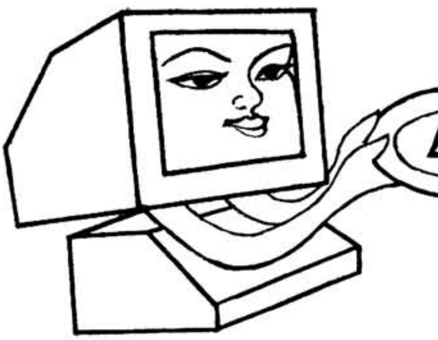
程序3 gtmono・exe程序

```
code    segment para public 'code'
        org 100h
main    proc far
        assume cs:code,es:code,ds:code
        jmp start
stack   segment para stack 'stack'
        db 64   dup(0)
stack   ends
start:  push ds
        mov ax,0
        push ax
        mov ax,cs
        mov es,ax
        mov ds,ax
        push es
        push ds
        xor ax,ax
        mov es,ax
        mov ax,es:[0410h]
        and al,0cfh
        or  al,30h
        mov es:[0410h],ax
        mov ax,0579h
        mov ds,ax
        mov dx,0da1h
        mov ax,2510h
        int 21h
        pop ds
        mov ax,2
        int 79h
        mov ah,0
        int 10h
        mov dx,0
        mov ah,2
        int 10h
        pop es
        ret
main    endp
code    ends
        end main
```

如果用dBASEⅢ做事务管理，则可以通过dBASEⅢ中的RUN命令将gtcolo・exe、graph・bas以及gtmono・exe等程序连接起来，特别有趣的是，当显示返回ADCH—DOS单色监视器工作后，系统将会继续运行dBASEⅢ命令程序，从而使整个事务管理系统产生图文并茂的效果。

四．结束语

以上我们介绍了在ADCH--DOS环境下实现IBM—PC双屏工作的一种方法，在实际工作过程中，我们切实体会到PC机使用双屏工作的确能给事务管理系统带来许多好处，特别是高分辨率的PC图形显示卡的出现，用户可以在彩色监视器上得到更高分辨率的彩色图形，因而也就使双屏工作更加有意义了。最后，需要指出的是，实现双屏工作，在不同的中文操作系统下，实现的方法是不完全相同的。本文只是介绍了在ADCH—DOS上实现双屏工作的一种方法。因而无论是对于ADCH—DOS还是其它的中文操作系统，本文所介绍的方法都不可能是唯一的，尽管如此，我们仍希望它能够为您的系统实现双屏工作起到抛砖引玉的作用。



一种菜单技术

西北工业大学　毛　军

用过d BASE Ⅱ及d BASE Ⅲ的人都知道，它们的命令程序的子程序不能嵌套四层，而对于一个比较大的系统来讲，这是很可能用到的，再者我们在用批命令去联接多种语言子程序的返回时用到的菜单也是一个棘手的问题。这里我们将分别提供一种比较方便的联接技术。

在d BASE Ⅲ中的设置函数功能，比如SET FUNCTION (n) TO'〈字符形表达式〉'。我们将用它把各个逻辑上的子程序联接起来，这里我们以d BASE Ⅲ为主进行介绍。

我们知道，当我们设置了SET FUNCTION 3 TO 'DO X1;'后，接着按下F3，机器将自动运行X1程序，如果我们将菜单方式的提示用F<n>表示，当用户选用F<k>时就相应地DO X<K>程序。当然运行一个程序是比较容易的，但需要解决的是用上述方式运行的程序结束后如何返回上一层程序的工作。因为F<n>并不是调用子程序，只是一种模拟，当然不会自动在运行后返回上一层菜单了，为了说明方便起见，我们举一个例子，该程序的主框图见图1，那么我们可利用下列程序去实现它们的联接，这里完全抛开了子程序的调用。

程序1：INT·PRG

```
CLEAR
SET FUNCTION 9 TO 'DO MAIN;'

SET FUNCTION 10 TO 'DO QUES;'

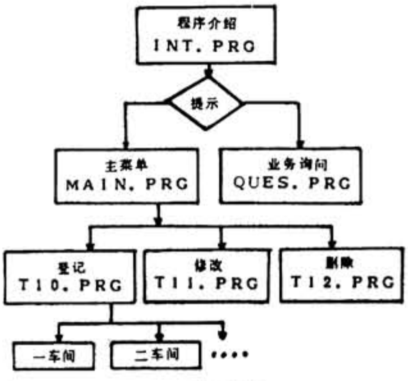
?   '目 录'
?   'F<9>……开始工作'
?   'F<10>……业务询问'
?   '请选用'
RETURN
```

程序2： MAIN·PRG

```
CLEAR
SET FUNCTION 3 TO 'DO T10;'
SET FUNCTION 4 TO 'DO T11;'
SET FUNCTION 5 TO 'DO T12;'
SET FUNCTION 10 TO 'DO INT;'
?      '目  录'
?      'F<3>……登  记'
?      'F<4>……修  改'
?      'F<5>……删  除'
?      'F<10>…退  出'
?      '请  选  用'
RETURN
```

程序3：T10·PRG

```
CLEAR
DO WHILE .T.
?  '   功能目录    '
?  '  0………退出   '
?  '  1………一车间  2……二车间   '
        .
        .
        .
ACCEPT '请选择' TO M
IF M=0
   DO MAIN
   RETURN
ENDIF
        .
        .
        .
ENDDO
```



图一　　主框图结构

这只是一个例子而已，它的内容并不重要，重要的是看看它们是如何接起来的。我们每一层选一个模块作了程序说明，同一层的返回情况是完全相同的。很显然，每个程序实际上已经是脱节的了，每次的提问都是在“· ”提示符下进行的，即此时已退出文件运行，但这无关紧要，只要逻辑上行得通即可。当然为了使整个系统更好一些，我们可将提示符“！”改为“请您选用”等字样：

程序4： CONFIG· DB

```
TALK=OFF
    .
    .
    .
    .
PROMPT='请您选用'
COMMAND=DO INT
    .
    .
    .
    .
```

在程序T10· PRG中，为了设置一个出口，故设立了“死循环”，以便只有当M=0时退出，这里的RETURN是不可少的，否则系统将会出现混乱，这一点请读者自已仔细考虑。

用类似的方法可以连接许多层程序，但在退出时必须有DO<上一层文件名>[回车] RETURN两条。（如果它是被F <n >调出的）。

现在我们用同样的方法去解决批处理的连接问题。类似地在子的程序的最后命令是上一层程序的文件名，比如下列程序：

程序5： ZK· BAT

```
ECHO OFF
DBASE ZK
```

程序6： ZK· PRG

```
ERASE
?    '目    录'
?    'A……程序7'
?    'B……程序8'
?    '  请选用  '
SET CONS OFF
QUIT
```

程序7： A· BAT

```
ECHO OFF
BASICA MX
ZK
```

（程序8—B。BAT与程序7结构相当，略去未写）。

置于ZK· PRG当然可用于其它的工作而不单单是一个菜单，BASICA的退出，只要在需要进入批文件的地方加上语句（number） SYSTEM[回车] 即可。

以上谈连接技术，请读者仔细体会，琢磨，了解整个连接的思想，以及整个运行过程，只有这样才能举一反三，在不同的場合下选用最佳方案

CWY系列交流参数稳压器技术交流订货会圆满结束

本刊讯　通讯员郑潮文报道：参加广州国营白云无线电厂第三届CWY系列交流参数稳压器技术交流订货会的单位及代表大部分是来自全国各地计算机行业的工程技术人员有计算机机房设计工程师、有计算机专家、计算机用户等。会上，冶金工业部秦皇岛黑色冶金矿山设计研究院计算机硬件工程师侯寿鹏同志对CWY系列交流参数稳压器给以高度评价。他所在单位负责计算机机房设计，为向计算机用户负责，他们对全国多数的稳压器进行测试比较，认为广州国营白云无线电厂生产的CWY系列交流参数稳压器各项技术性能比其它厂家好，且符合配置计算机的要求。所以，该研究院为各单位计算机房设计时都选用CWY系列交流参数稳压器。广西计委计算站动力设备科王昌明同志介绍了该区电压低，计算机运行不正常，经过充分比较之后，他们去年购进了一批CWY系列交流稳压器，并对该稳压器高度的评价。会议期间，参加技术交流的单位、专家代表之间作了技术交流，收获很大，很多同志希望明年继续召开及扩大这样的技术交流订货会。

CP /M工具软件ED程序剖析(四) ED子命令执行原理

中山大学计算机系 郭嵩山 曾超荣 郑吉丰

一．ED程序子命令功能分类

ED程序有25条子命令，子命令的执行过程与其功能有着密切的联系，同类型功能的命令，其执行过程往往有许多相同或相似的地方，为了便于读者理解，我们把ED子命令从功能角度划分，大致可以分成如下五种类型。

1·文件编辑作业类

这类子命令主要用于正常或非正常结束文件编辑，属于这类子命令有E(结束编辑，返回CP/M系统)、H(结束编辑、重新进入新的编辑状态)、O(废除原先编辑作业，重新进入新的编辑状态)、Q(废除原先的编辑作业，返回CP/M系统)。

2·读写插入类

这类命令主要功能是在缓冲区内进行读、写、插入字符的操作，具体的又根据所读入文件类型分作两种：

⑴一般文件读、写插入类：这类命令是读人的源文件和生成的文件都是非库文件，属于这类命令的有A(添加文本)、W(写行)、I(插入字符)。

⑵库文件读写类：属于这类子命令有R(读库文件到缓冲区CP后)、X(写系统所提供的X.$$$$$$$·LIB库文件)。

3·编辑缓冲区操作类

属于这类的子命令较多，细分还可分为：

⑴缓冲区字符指针操作类：属于这类子命令有B(CP移到缓冲区开头或末尾)、C(移字符)、L(移行)、D(删字符)、K(删行)。

⑵显示(打印)缓冲区内容类：属于这类子命令有P(显页)、T(显行)、±n(移n行后打印当前行)。

⑶检索替换字符串类：属于这类子命令有F(寻找字符串)、J(置字符串于指定位置)、S(替换字符)、N(寻找缓冲区内或源文件上的内容)。

4·宏命令操作类

属于这类命令就是宏命令(M)，它是一个重复执行的命令，可以将多条ED子命令组合成命令串进行操作。

5·实用功能类

这类命令本身并不直接进行编辑作业，而是为完成一些特定功能而设。属于这类命令的有U(大小写转换)、V(显示缓冲区空间大小或显示行号)、Z(延时等待)。

下面举一些子命令执行的典型例子，来说明ED程序是如何实现子命令功能的。

二．文件编辑作业类子命令的执行

图4·1，图4·2分别示出了E、Q命令执行的流程。

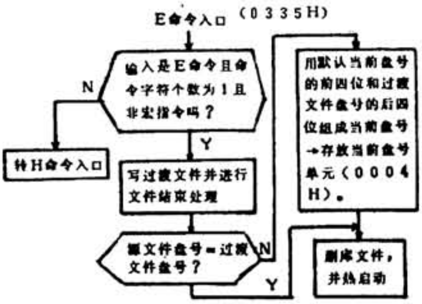


图4.1 E命令执行过程

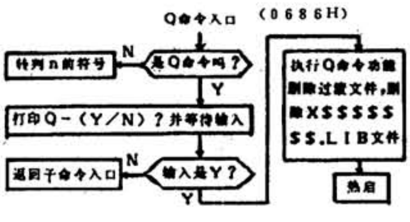


图4.2 Q命令执行过程

在图4·1中，写过渡文件并进行文件结束处理主要是执行首址为15BD的子程序来完成，其

具体功能是将编辑缓冲区的内容及尚未进行处理的源文件缓冲区内容抄录到过渡文件中，然后关闭文件及进行文件改名，即将原来的源文件改名为备份文件，过渡文件改名为新的源文件。

H、O命令执行流程与E、Q相似，但却增加了一条返回执行编辑预处理子程序的指令（编辑预处理子程序流程请参看今年第二期的图2·2），以能在结束编辑（H命令）或废除本次编辑作业（O命令）之后能重新进入编辑状态。此外，由于重新编辑作业是在源盘上进行的，所以H命令在进行盘号处理时，将过渡文件盘号与源文件盘号互换。

三、读写插入类子命令的执行

1·一般文件读写插入类子命令的执行。

A（添加文本）和W（写行）命令读写时缓冲区指针移动原理已在上期介绍，图4·3，图4·4分别示出了其执行的流程。

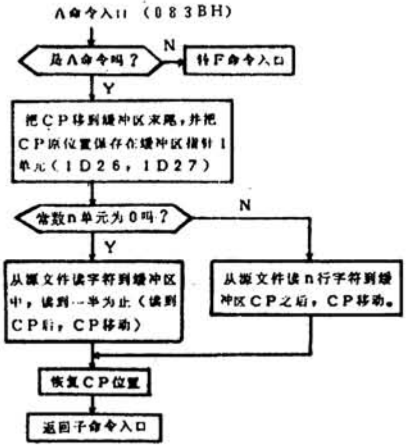


图4．3 A命令执行过程

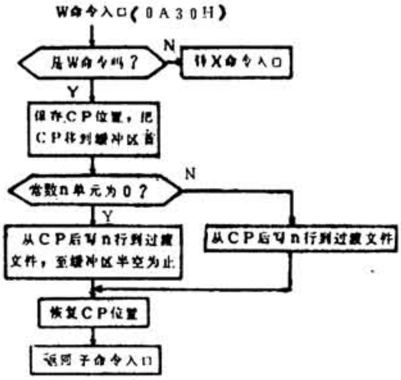


图4．4 W命令执行过程

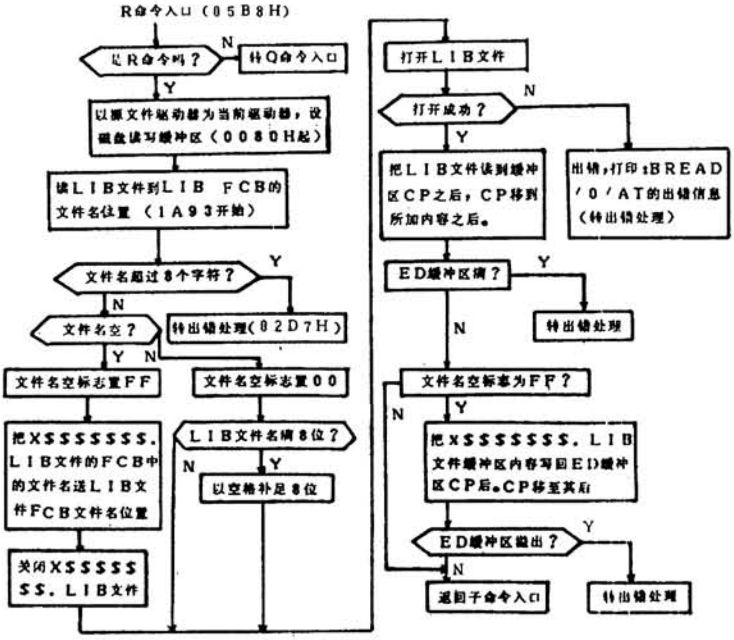


图4．5 R命令执行过程

I 命令是在E D状态下插入字符，其执行过程是：在进行插入预处理后，从控制台读入一字符，并判所读入的字符是否↑Z、↑U、↑X、↑R、↑H、DEL、FF和TAB，如果是则进行相应的处理，如果不是，就将所输入的字符存入编辑缓冲区，并对输入的CR、LF(回车、换行)作相应处理。

2．库文件读写类子命令的执行

R命令功能是将库文件读到编辑缓冲区CP之后，如果库文件空，则以E D程序本身设定的X $ $ $ $ $ $ $·LIB为库文件，库文件插入后，CP移到所插入内容之后，其流程如图4·5 所示。

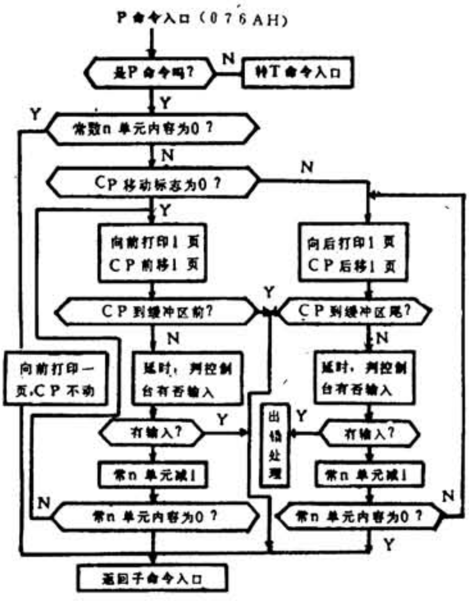


图4·6 P命令执行过程

X命令是用于写设定的X $ $ $ $ $ $ $·LIB库文件，其命令形式是n X，当n ≠0 时，写入n 行；当n =0 则转作删除X $ $ $ $ $ $ $·LIB文件处理，而且X $ $ $ $ $ $ $·LIB库文件只能写一次，如写第二次则会将原来内容全部删去，该库文件仅存在于编辑过程中，编辑结束时(执行E、H、Q、O、命令)，该库文件会自动消失。

四．编辑缓冲区子命令的执行

1．缓冲区字符指针操作类子命令的执行。

B、C、L、D、K命令执行比较简单，B、C 主要调用了编辑缓冲区两端字符间传送子程序(首址为1452)，D、K调用了编辑缓冲区当前字符指针移动子程序(首址为1458)，L调用了按行移动编辑缓冲区字符指针子程序(首址为145E)。为节省篇幅，在此就不详述了。

2．显示(打印)类子命令的执行。

P 命令是按页打印命令，图4·6 示出了其流程图，如图4·6 所示，若n =0，则向前打印一页，CP不动；否则向前(－nP)或向后(nP)打印n 页，打完一页后CP移动，并延时，判控制台有输入否，如果有输入，则作出错处理，否则继续执行。T命令的功能是向前或向后显示n 行，CP不动，其执行比较简单，主要调用了首址为1746 显示缓冲区内容的子程序。±n 命令功能实际

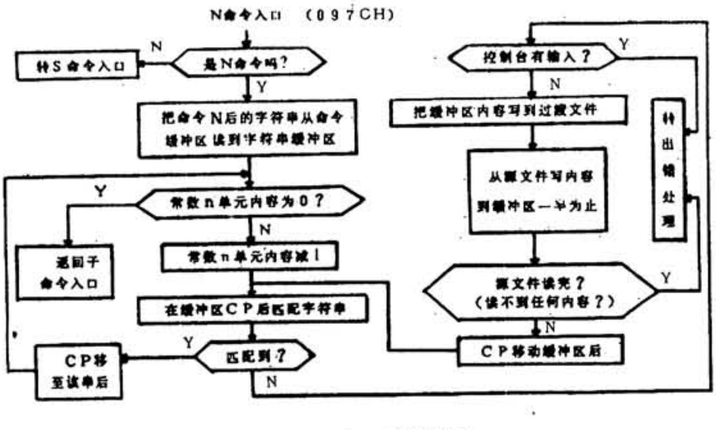


图4.7 N命令执行过程

上是执行组合的±n LT命令，其实现也较简单，在此不再详述。

3. 检索替换字符串类子命令的执行

F、N、J、S都是检索字符串的命令，命令前的n均表示要查找的次数。下面以N、J命令执行为例进行说明。

图4.7，图4.8分别示出了N、J命令执行的流程。

在图4.7中，执行N命令，从缓冲区CP后检索字符串，如完成查找(n次)，则将CP移到编辑缓冲区中该串之后；否则，继续把源文件的内容读到编辑缓冲区，进行查找。如果源文件读完后仍没找到该字符串(n次)，则发出错信息。

在图4.8中，执行J命令，J命令后跟三串字符，执行时先在缓冲区CP后查找第一串字符，找到后将第二串字符插入其后，CP移至所插入字符末尾；再查找第三串字符，找到后将此串前(不包括该串)到CP后的所有字符删去。执行J命令，正常结束返回子命令入口，如未能匹配到，则转出错处理。

F命令与S命令与N、J命令执行较类似，但比较简单。F命令的功能是在编辑缓冲区CP之后查找字符串，查到后CP移到该字串末尾。实际上，F命令执行了N命令的前半部分(在缓冲区内查找字符串)。S命令的执行也与J命令前半部分相似，在找到第一串字符后，以第二串字符取代第一串字符，CP移到替换字符之后。

五、宏命令的执行

宏命令(M)是将ED子命令组合起来，重复执行。其命令格式是：

$$n\,M\,d_1\,d_2\,d_3\cdots d_n$$

其中n表示执行次数，$d_1d_2\cdots d_n$表示命令串。M命令的执行，关键是从何处读取命令，在《ED程序编辑的预处理和子命令执行总体流程》一文中，我们已介绍了ED是如何读取命令的。M命令的处理程序，是通过将命令缓冲区M后的命令串搬至M命令缓冲区中，并进行一些必要的处理，图4.9示出了M命令处理的流程。

六、实用功能类子命令的执行

这类子命令执行较简单，下面只示出V命令执行流程(图4.10)，其余就不详述了。

至此，我们已分四期分析完ED程序的结构。由于篇幅所限，有些细节未能详谈，但读者已完全可以根据以上分析，较易地阅读ED程序。由于笔者水平所限，有不当之处，欢迎指正

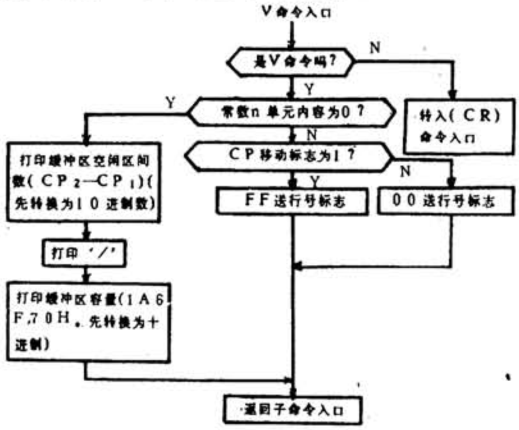


图4.10 V命令执行过程

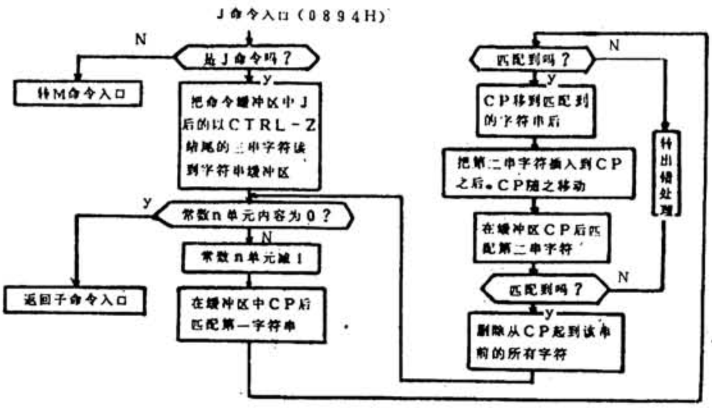


图4.8 J命令执行过程

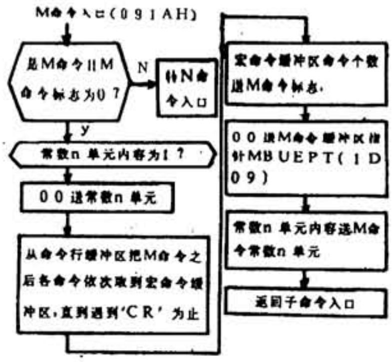


图4.9 M命令执行过程

APPLEⅡ 低分辨率直方图显示及其打印技巧
——显示和打印实用柱状统计图表的方法

广州大学计算中心 黄伟民

在用APPLE—Ⅱ机作管理或进行数据处理时，常常要用到分类统计，通常大多在文本状态下用按一定比例打印一行行“*”号来解决，这虽然可以，但不够美观。用直方图（柱状图）显示的分类统计图表很漂亮，但一般打印不出来，原因是APPLE—Ⅱ只有高分辨率图形打印功能，而没有低分辨率图像打印功能。

笔者最近编制的程序（以一个班某次考试成绩分类统计为例）可以实现对一系列数据进行统计分类，按一定比例显示分类统计直方图重新将其所显示的直方图打印出来，图形非常直观、漂亮。稍加修改，可用到其他各类分类统计报表中，如试验数据分析、销售统计、抽样调查、生产统计报表等。进一步修改还可打印出对比统计图表等，很有实用价值。

下面是该程序清单及所打印出的低分辨率直方图。

程序2500—2560行显示按程序中数据（此处为49个，运行时应输入49）分类统计的低分辨率直方图。

程序2561—2569行可打印出所显示的直方图（该图形是用CP—80打印机打印出来的。如用其他型号的打印机，可将2566行中之CHR$（16）改为CHR$（35）（即为“#”号）或其他适当字符打印）。

2561——2569行如用于其他所显示低分辨率图像程序后面，便可将所显示其他低分辨率图像打印出来。读者不妨一试。

```
]LIST

10  REM   THE PROGRAM FOR TO PRIN
    T [GR]-HISTOGRAM
30 D$ =  CHR$ (4)
40  PRINT D$;"PR#1"
45  INPUT "N=?";N
46  PRINT : PRINT : PRINT
48 K = N
50  DIM A(49),X(49)
100  FOR I = 1 TO K
110  READ A(I)
120  NEXT I
130  REM  UNDERNEATH IS N DATA
150  DATA  69.6,93,58.4,74.9,61.6
    ,87.7,69.2,80.4,66.6,78.33,6
    2.25,55.9,90.6,58.5
160  DATA  58.74,69.6,80.4,82.24,
    85.9,62.2,79.3,73.04,68.1,10
    0,52.3,86.7,74.14,77.24,68.3
    4
170  DATA  77.9,60.7,43.3,44.5,66
    .3,46.28,54.5,47.1,0,100,74.
    5,66.4,62.5,73.74,60.4,73.6,
    54.8,79.1,77.94,69.2
800  FOR I = 1 TO K
803 X(I) = A(I)
805  NEXT
807  FOR I = 8 TO 0 STEP  - 1
808 Y%(I) = 0
809  NEXT I
810  GOSUB 2000
820  GOSUB 2500
825  PRINT D$;"PR#0"
830  END
2000  HOME
2010  FOR I = 1 TO K
2020  IF X(I) < 30 THEN Y%(0) = Y
    %(0) + 1: GOTO 2040
2030 H = X(I) / 10 - 2:Y%(H) = Y%
    (H) + 1
2040  NEXT I
2050 M = Y%(0)
2060  FOR I = 1 TO 8
2070  IF M < Y%(I) THEN M = Y%(I)

2080  NEXT
2090  RETURN
2500  GR
2510  COLOR= 15
2520  VLIN 39,39 - Y%(8) * 39 / M
    AT 2
2525 N = 0
2530  FOR I = 7 TO 0 STEP  - 1
2540 N = N + 1
2550  VLIN 39,39 - Y%(I) * 39 / M
    AT 2 + 4 * N
2560  NEXT I
2561 D$ =  CHR$ (4): PRINT D$;"PR
    #1": POKE 1657,255
2562  PRINT  CHR$ (27); CHR$ (65)
    ; CHR$ (8);
2563  FOR J = 0 TO 39
2564  FOR I = 0 TO 39
```

APPLE Ⅱ高分辨图像的缩制

陈杰林

在设计APPLE　SOFT程序时，如果程序要用到高分辨图像作底图，且图像比较复杂，以往办法是：先把底图在机上用程序绘好，然后把整页图像存贮区的数据当作机器码数据的形式存到磁盘去，当程序要用到该页图像时，再用BLOAD命令把它装回存贮区。这个办法虽然比较简单，但占用磁盘段落较多，使到一张磁盘存放图像的幅数不多。针对这个问题，笔者设计了一个专门程序，对已描绘好的高分辨图像进行处理，经过处理的图像变成机器码直接程序的形式存到磁盘。使用的时候直接用BRUN命令即可还原，或用BLOAD命令装入，再用CALL 28533命令调用。经过处理再存放的一页图像，能够节省磁盘段落1／3～2／3（视图像的复杂程度），对图像的彩色效果没有影响，大大提高了磁盘的使用效率。

考虑到处理速度的问题，本程序的主要部分用汇编语言来设计，另外再用一段BASIC程序来调用汇编程序。所以要先在磁盘建立这二段机器码子程序。程序不算太长，可用小汇编办法键入即可，下面是程序清单：

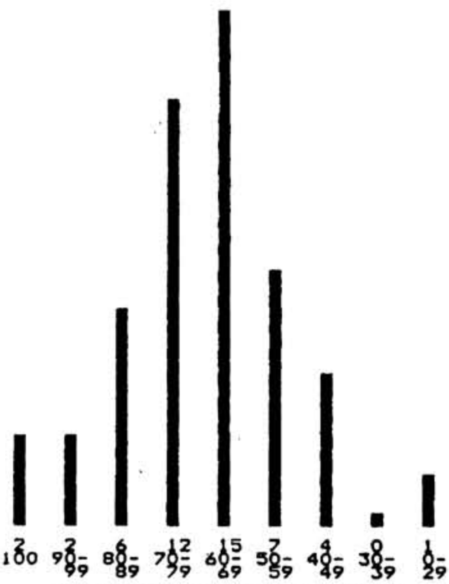
［程序1］

```
6000-   D8          CLD
6001-   AD AC 60    LDA   $60AC
6004-   85 1B       STA   $1B
6006-   AD AD 60    LDA   $60AD
6009-   85 1C       STA   $1C
600B-   AD AE 60    LDA   $60AE
600E-   85 19       STA   $19
6010-   AD AF 60    LDA   $60AF
6013-   85 1A       STA   $1A
6015-   A2 00       LDX   #$00
6017-   AD B0 60    LDA   $60B0
601A-   C5 19       CMP   $19
601C-   D0 08       BNE   $6026
601E-   AD B1 60    LDA   $60B1
6021-   C5 1A       CMP   $1A
```

```
2565  IF  SCRN( I,J) < 1 THEN  PRINT
     " ";: GOTO 2567
2566  PRINT  CHR$ (16);
2567  NEXT I
2568  PRINT
2569  NEXT J
2570  HOME
2575  PRINT
2580  HTAB 3: PRINT Y%(8);
2590 N = 0
2600  FOR I = 7 TO 0 STEP  - 1
2610 N = N + 1
2620  HTAB 3 + 4 * N: PRINT Y%(I)
     ;
2630  NEXT I: PRINT
2640  INVERSE
2650  PRINT  TAB( 2)100;
2660  FOR I = 7 TO 1 STEP  - 1
2670  PRINT " ";10 * (I + 2);"-";

2680  NEXT I: PRINT  TAB( 35)0;"-
     "
2690  HTAB 5
2700  FOR I = 7 TO 0 STEP  - 1
2710  PRINT "  ";10 * (I + 3) - 1
     ;
2720  NEXT I: PRINT
2730  FOR I = 1 TO 30000: NEXT I
2740  NORMAL
2745  PRINT : PRINT  TAB( 6)"CLAS
     SIFIED-STATISTICS GRAPH"
2748  PRINT : PRINT
2750  TEXT : HOME
2756  PRINT  CHR$ (27); CHR$ (65)
     ; CHR$ (13)
2758  PRINT
2760  RETURN

]RUN
N=?49
```



CLASSIFIED-STATISTICS GRAPH

```
6023-   D0 01       BNE  $6026
6025-   60          RTS
6026-   A0 00       LDY  #$00
6028-   B1 19       LDA  ($19),Y
602A-   C9 7F       CMP  #$7F
602C-   F0 07       BEQ  $6035
602E-   C9 00       CMP  #$00
6030-   F0 0A       BEQ  $603C
6032-   4C 83 60    JMP  $6083
6035-   A9 7F       LDA  #$7F
6037-   85 1F       STA  $1F
6039-   4C 40 60    JMP  $6040
603C-   A9 00       LDA  #$00
603E-   85 1F       STA  $1F
6040-   E0 00       CPX  #$00
6042-   D0 20       BNE  $6064
6044-   A0 01       LDY  #$01
6046-   B1 19       LDA  ($19),Y
6048-   C5 1F       CMP  $1F
604A-   D0 37       BNE  $6083
604C-   C8          INY
604D-   B1 19       LDA  ($19),Y
604F-   C5 1F       CMP  $1F
6051-   D0 30       BNE  $6083
6053-   18          CLC
6054-   A5 1B       LDA  $1B
6056-   85 1D       STA  $1D
6058-   69 02       ADC  #$02
605A-   85 1B       STA  $1B
605C-   B0 44       BCS  $60A2
605E-   A5 1C       LDA  $1C
6060-   85 1E       STA  $1E
6062-   A0 00       LDY  #$00
6064-   A5 1F       LDA  $1F
6066-   09 80       ORA  #$80
6068-   91 1D       STA  ($1D),Y
606A-   E8          INX
606B-   F0 A8       BEQ  $6015
606D-   8A          TXA
606E-   A0 01       LDY  #$01
6070-   91 1D       STA  ($1D),Y
6072-   E6 19       INC  $19
6074-   D0 02       BNE  $6078
6076-   E6 1A       INC  $1A
6078-   A0 00       LDY  #$00
607A-   B1 19       LDA  ($19),Y
607C-   C5 1F       CMP  $1F
607E-   D0 95       BNE  $6015
6080-   4C 17 60    JMP  $6017
6083-   A0 00       LDY  #$00
6085-   B1 19       LDA  ($19),Y
6087-   C9 FF       CMP  #$FF
6089-   F0 04       BEQ  $608F
608B-   C9 80       CMP  #$80
608D-   D0 02       BNE  $6091
608F-   29 7F       AND  #$7F
6091-   91 1B       STA  ($1B),Y
6093-   E6 1B       INC  $1B
6095-   D0 02       BNE  $6099
6097-   E6 1C       INC  $1C
6099-   E6 19       INC  $19
609B-   D0 02       BNE  $609F
609D-   E6 1A       INC  $1A
609F-   4C 15 60    JMP  $6015
60A2-   18          CLC
60A3-   A5 1C       LDA  $1C
60A5-   85 1E       STA  $1E
60A7-   E6 1C       INC  $1C
60A9-   4C 64 60    JMP  $6064
60AC-   00          BRK
60AD-   70 00       BVS  $60AF
60AF-   20 00 40    JSR  $4000
60B2-   00          BRK
60B3-   00          BRK
```

以上程序键入完毕后，打入“BSAVE GRAPE P.S1，A$6000，L$B2”[回车] 将程序1 命名为“GRAPE P.S1”录进磁盘。然后继续用小汇编的办法输入程序2 。程序2 清单如下：

[程序2]

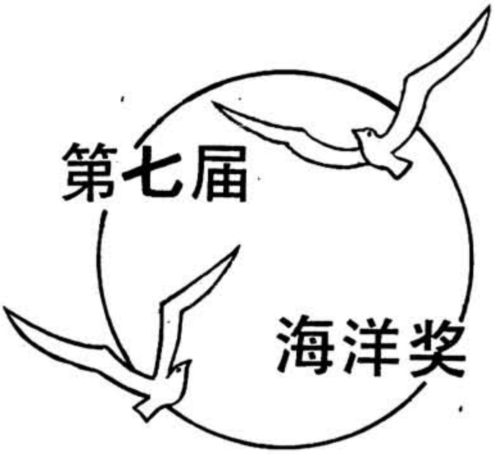
```
6F75-   D8          CLD
6F76-   AD FA 6F    LDA  $6FFA
6F79-   85 1B       STA  $1B
6F7B-   AD FB 6F    LDA  $6FFB
6F7E-   85 1C       STA  $1C
6F80-   AD FE 6F    LDA  $6FFE
6F83-   85 19       STA  $19
6F85-   AD FF 6F    LDA  $6FFF
6F88-   85 1A       STA  $1A
6F8A-   A0 00       LDY  #$00
6F8C-   B1 19       LDA  ($19),Y
6F8E-   C9 80       CMP  #$80
6F90-   F0 07       BEQ  $6F99
6F92-   C9 FF       CMP  #$FF
6F94-   F0 03       BEQ  $6F99
6F96-   4C CE 6F    JMP  $6FCE
6F99-   A0 01       LDY  #$01
6F9B-   B1 19       LDA  ($19),Y
6F9D-   8D 72 6F    STA  $6F72
6FA0-   A0 00       LDY  #$00
6FA2-   B1 19       LDA  ($19),Y
6FA4-   29 7F       AND  #$7F
6FA6-   A0 00       LDY  #$00
6FA8-   91 1B       STA  ($1B),Y
6FAA-   C8          INY
6FAB-   CC 72 6F    CPY  $6F72
6FAE-   90 F8       BCC  $6FA8
6FB0-   18          CLC
6FB1-   A5 19       LDA  $19
6FB3-   69 02       ADC  #$02
6FB5-   85 19       STA  $19
6FB7-   A5 1A       LDA  $1A
6FB9-   69 00       ADC  #$00
6FBB-   85 1A       STA  $1A
6FBD-   18          CLC
6FBE-   A5 1B       LDA  $1B
6FC0-   6D 72 6F    ADC  $6F72
6FC3-   85 1B       STA  $1B
6FC5-   A5 1C       LDA  $1C
6FC7-   69 00       ADC  #$00
6FC9-   85 1C       STA  $1C
6FCB-   4C E0 6F    JMP  $6FE0
6FCE-   A0 00       LDY  #$00
6FD0-   B1 19       LDA  ($19),Y
6FD2-   91 1B       STA  ($1B),Y
6FD4-   E6 19       INC  $19
6FD6-   D0 02       BNE  $6FDA
6FD8-   E6 1A       INC  $1A
6FDA-   E6 1B       INC  $1B
6FDC-   D0 02       BNE  $6FE0
6FDE-   E6 1C       INC  $1C
6FE0-   38          SEC
6FE1-   AD FC 6F    LDA  $6FFC
6FE4-   E5 1B       SBC  $1B
6FE6-   AD FD 6F    LDA  $6FFD
6FE9-   E5 1C       SBC  $1C
6FEB-   B0 9D       BCS  $6F8A
6FED-   AD 50 C0    LDA  $C050
6FF0-   AD 57 C0    LDA  $C057
6FF3-   AD 52 C0    LDA  $C052
6FF6-   AD 54 C0    LDA  $C054
6FF9-   60          RTS
6FFA-   00          BRK
6FFB-   20 FF 3F    JSR  $3FFF
6FFE-   00          BRK
6FFF-   70 FF       BVS  $7000
7001-   FF          ???
```

程序编好后，打入“BSAVE GRAPE P.S2，A$6F75，L$8B”[回车]，把程序录进磁盘。以上两程序编好并录进磁盘后，就可以执行以下这个BASIC程序处理你所需要的图像了。这个程序使用方法比较简单，执行之后程序先从磁盘调入机器码程序，然后根据你的图像所在的存贮区输入1 或2 ，再输入你的图像录往磁盘用的名字就可以完成一幅图像的加工了。下面是程序清单：

```
]LIST

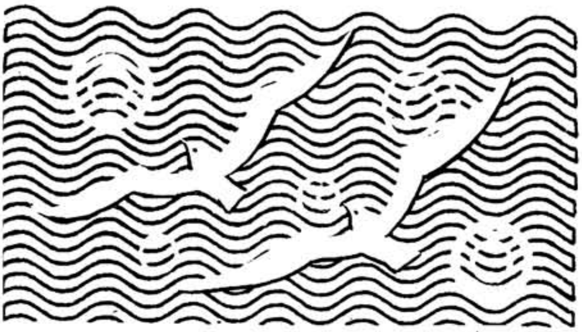
10 D$ =  CHR$ (4)
20  PRINT D$;"BLOAD GRAPE P.S1"
30  PRINT D$;"BLOAD GRAPE P.S2"
40  INPUT "HGR OR HGR2 (1/2)?";A
50  IF A < 1 OR A > 2 THEN 40
60  INPUT "NAME:";NAME$
70  IF A = 2 THEN 90
80  POKE 24751,32: POKE 24753,64:
     POKE 28663,84: POKE 28667,3
    2: POKE 28669,63
85  GOTO 100
90  POKE 24751,64: POKE 24753,96:
     POKE 28663,85: POKE 28667,6
    4: POKE 28669,95
100  CALL 24576
110  PRINT D$;"BSAVE";NAME$;",A$6
     F75,L";( PEEK (28) - 112) *
     256 +  PEEK (27) + 140
120  END

]
]RUN
HGR OR HGR2 (1/2)?1
NAME:ABC
```



第七届 海洋奖

入选题目及作者



(1) 有2 n 个人买电影票，五角一张，其个n 个人持五毛钱，n 个人持一元钱，要求计算机打印出所有排列方法，使这2 n 个人买票期间不出现找不开的现象。

(北京大学附中高一(3)　毕志强)

(2) 600！能被9 的最高多少次方整除?

(杭州武警指挥学校文化教研室　任　鸣)

(3) 编程模拟出“深邃的夜空中，繁星密布，小星星眨着眼睛，时隐时现，在玩捉迷藏”的景象。要求：用一行程序实现。

(甘肃兰州一中高二(3) 班　王　森)

(4) 请编一简单程序，可以把十进制数和十六进制数进行互相转化。

(江四省赣州市一中　张启亭)

(5)有十五个黑人和十五个白人，将他们排成一个圆圈。从一人开始数起，数到十就叫此人退出圈子，取消继续资格，然后重新数，当去掉十四人后，黑人只剩下一个了，于是改变方法，从这一个黑人开始数起，以相反方向去数，仍然数到十便叫出圈子，最后又只剩下了这一个黑人，问当初是怎样排列的?

(河南省湘潭市一中199班　曹　洋)

(6) 在N个连成一串的方格内填入“R”和“B”，但相邻两格内不能都填“R”，编程序，输入N，打印出所有填法和总方案数。

例如在4 个方格内，下图为一填法：

R	B	B	R

(浙江省宁波市效实中学高二(2) 班　徐坚雄)

(7) 三个相邻偶数的乘积等于87…………8，试编一程序求出这些数。

(浙江省台州中学高二(3) 班　石　京)

(8) 编一程序，从键盘输入三角形三边长，若输入三边长的值不能构成三角形显示“*”，对于能构成三角形的：等边三角形显示“3”，等腰三角形则显示“2”，否则显示“0”。对于能构成三角形的，还要打印显示其周长和面积(要求不多于两行程序)。

(广西北海市北海中学高一(1) 班　陈兆前)

(9) 编一个一行程序，找出输入的b 个数的最大数和最小数。

(广州市实验学校高二(1) 班　梅　坚)

以上每道题目作者均奖励海洋牌双面双密度电脑软磁盘一片(奖品由深圳市南头较场后海洋音响器材总公司深圳分公司电脑磁盘厂资助。该厂生产电脑软磁盘、录音带、原声带、空白磁带等。电话：23711转45342，电挂：0742)。

本刊87年8期将公布第七届题目征答入选者名单及答案选登，87年5 期将公布第八届海洋奖入选题目，第6 期公布第五届征答入选作者名单及征答选，欢迎踊跃参加征题征答活动，寄来的题目答案均入选者奖励电脑软磁盘2 片。来稿请于信封上面写上“征题征答”字样。

本刊连续12届开展有奖征题征答，每届奖励20名，征题征答对象均不限。

寄来的题目(要求自编) 请提供参考程序。程序及运行结果请打印在同一张纸上。

每届全部题目都答或选答均可。每道程序及运行结果均占一张纸，每张纸上面均写“×届×题答案”，纸下方写上作者姓名及详细地址。

电脑辅助抛体运动的教学

安徽省定远第一中学物理组　肖学璞　吴义琴

抛体运动的规律是中学物理教学重要课题之一。对于弹道曲线细究一下，会看到有不少的有趣的规律。用电脑辅助这一教学，更使师生乐在其中，有益于教与学。

在不考虑空气阻力，于二维空间里，弹导曲线的规律有：

一、初速度相同：45°的抛射角时射程最大；抛射角互余，射程相同。（见图一）

二、抛射角相同，抛出速度大小不同，轨道的最高点在一条通过抛出点的直线上。（见图二）

三、抛射速度大小相等，抛射角不同，各轨道最高点在一个和抛出面相切于抛出点的椭圆上。（见图三）

四、以相同大小的抛射速度，向不同方向抛出的物体都处于某一条曲线——包络线即所谓的“火力圈”内。（见图四）

五、由同一点以同样大小的速度向各个方向同时抛出若干个物体，这些物体在任一时刻（未落地之前）均处于某同一个圆周上。（见图五）

六、以相同的抛射角不同大小的抛射速度同时抛出若干个物体，这些物体（未落地前）任何一时刻均位于同一直线上。（见图六）

我们将以上规律编成程序在APPLE Ⅱ上运行，显示效果见六幅硬拷贝，并附程序于后。

本程序使用说明：

一、N是抛体个数或次数。

10——40行绘制坐标。

70——180行描绘图一中的轨道曲线。

200——420行描绘图二中的直线。

500——550行描绘图三中的椭圆。

585——640行描绘图四中的“火力圈”。

700——900行描绘图五中的圆周。

1000——1200行描绘图六中的平行线族。

二、运行中据屏中显示的要求输入角度，速度值（如果过大会自动要求重输）。

据显示屏显示的50行，300行，570行，896行，1175行的A，B，C，D，E的要求，键入适当的数，这时便会得到相应的图线。具体是：

A＝1→B＝1→绘图一曲线。

A＝1→B＝2→绘图二直线。

A＝1→B＝3→绘图三椭圆；

B＝3→C＝1 保留椭圆；

C＝2 擦去椭圆；

C＝2 擦去椭圆；

C＝3 绘图四“火力圈”。

A＝2→绘图五圆，D＝1→继续下一个；

D＝2 停止。

A＝3→绘图六平行斜线；

E＝1→继续下一个；E＝2 停止。

三、附有关曲线的方程：

1．图一中的弹道线方程：

$$h=\tan\theta\cdot s-\frac{g}{2v_0{}^2\cos^2\theta}\cdot S^2$$

2．图二中的直线方程：

$$h=\frac{1}{2}\tan\theta\cdot s$$

3．图三中的椭圆方程：

$$\frac{(s-0)^2}{(v_0{}^2/2g)^2}+\frac{(h-v_0{}^2/4g)^2}{(v_0{}^2/4g)^2}=1$$

4．图四中的“火力圈”方程：

$$h=v_0{}^2/2g-g/2v_0{}^2\cdot s^2$$

5．图五中的圆方程：$(s-0)^2+[h-(-1/2gt^2)]^2=(v_0t)^2$。

```
ULIST,58Ø

3  PRINT  TAB( 1Ø);"****----HELLO
     ----****"
8  PR# 1: POKE 1913,1
1Ø  HGR : HCOLOR= 7
12  INPUT "N=";N
15  DIM A(N),V(N),H(N),R(N),S(N),
     T(N)
2Ø  HPLOT 13Ø,158 TO 13Ø,2 TO 127
     ,7: HPLOT 133,3 TO 133,1Ø: HPLOT
     137,3 TO 137,1Ø: HPLOT 134,7
     TO 136,7
3Ø  HPLOT 2,8Ø TO 265,8Ø TO 26Ø,7
     7: HPLOT 263,83 TO 257,83 TO
     257,86 TO 263,86 TO 263,89 TO
     257,89
4Ø  HPLOT 128,82 TO 125,82 TO 125
     ,86 TO 128,86 TO 128,82
5Ø  INPUT "A=?1,?2,?3=";A
51  IF A = 1 THEN 7Ø
52  IF A = 2 THEN 7ØØ
53  IF A = 3 THEN 1ØØØ
7Ø  FOR I = 1 TO N
72  PRINT "A(";I;")=";
73  INPUT A(I)
79  PRINT "V(";I;")=";
8Ø  INPUT V(I)
81 R(I) = V(I) * V(I) *  SIN (2 *
     Ø.Ø1745 * A(I)) / (2 * 9.8)
82 H(I) = V(I) * V(I) *  SIN (Ø.Ø
     1745 * A(I)) *  SIN (Ø.Ø1745
     * A(I)) / (2 * 9.8)
83  IF H(I) > 8Ø OR 2 * R(I) > 14
     5 THEN 72
9Ø T(I) = 2 * V(I) *  SIN (Ø.Ø174
     5 * A(I)) / 9.8
95 T(I) =  ABS (T(I)):R(I) =  ABS
     (R(I)):H(I) =  ABS (H(I))
96 H(I) =  INT (H(I) * 1ØØØ) / 1Ø
     ØØ:R(I) =  INT (R(I) * 1ØØØ)
     / 1ØØØ:T(I) =  INT (T(I) *
     1ØØØ) / 1ØØØ
98  PRINT "H(";I;")=";H(I); SPC(
     2);"S(";I;")=";2 * R(I); SPC(
     2);"T(";I;")=";T(I)
1ØØ  FOR T = Ø TO T(I) STEP T(I) /
     5Ø
11Ø X = V(I) *  COS (Ø.Ø1745 * A(
     I)) * T
12Ø Y = V(I) *  SIN (Ø.Ø1745 * A(
     I)) * T - 9.8 * T * T / 2
13Ø  CALL 64477
14Ø  HPLOT 13Ø + X,8Ø - Y
17Ø  NEXT T
18Ø  NEXT I
22Ø  FOR I = 1 TO N - 1
23Ø  FOR J = I + 1 TO N
24Ø  IF H(I) > H(J) THEN H = H(I)
     :H(I) = H(J):H(J) = H:R = R(
     I):R(I) = R(J):R(J) = R
25Ø  NEXT J
26Ø  NEXT I
3ØØ  INPUT "B=?1,?2,?3=";B
31Ø  IF B = 1 THEN 42Ø
311  IF B = 2 THEN 4ØØ
312  IF B = 3 THEN 5ØØ
4ØØ  HCOLOR= 3
41Ø  HPLOT 13Ø,8Ø TO 13Ø + R(N),8
     Ø - H(N)
42Ø  END
5ØØ  HCOLOR= 3
5Ø5 H1 = V(N) * V(N) / (2 * 9.8)
51Ø  FOR Y = Ø TO H1 STEP H1 / 1Ø
     Ø
52Ø X =  SQR ((1 - (Y - H1 / 2) *
     (Y - H1 / 2) / (H1 * H1 / 4)
     ) * (H1 * H1))
53Ø  HPLOT 13Ø + X,8Ø - Y
54Ø  HPLOT 13Ø - X,8Ø - Y
55Ø  NEXT Y
57Ø  INPUT "C=?1,?2,?3=";C
58Ø  IF C = 1 THEN 64Ø

ULIST582,111Ø

582  IF C = 2 THEN  HCOLOR= Ø: GOTO
     51Ø
585  HCOLOR= 3
6ØØ  FOR X =  - V(N) * V(N) / 9.8
     TO V(N) * V(N) / 9.8 STEP V
     (N) * V(N) / (9.8 * 8Ø)
61Ø Y = H1 - X * X / (4 * H1)
62Ø  HPLOT 13Ø + X,8Ø - Y
63Ø  NEXT X
64Ø  END
7ØØ  HPLOT 128,8Ø TO 128,3Ø TO 13
     2,3Ø TO 132,8Ø
7Ø5  HCOLOR= 3
71Ø SC = 1.1
72Ø CX = 13Ø:CY = 3Ø
73Ø  INPUT "VØ=";VØ
74Ø  FOR I = 1.Ø TO 4.Ø
75Ø  FOR T = I - 1.Ø TO I STEP Ø.
     Ø2
755 IT = Ø
76Ø  FOR D = Ø TO 2 * 3.1415926 STEP
     2 * 3.1415926 / N
77Ø X = VØ *  COS (D) * T:Y = VØ *
     SIN (D) * T - 9.8 * T * T /
     2
78Ø SX = SC * X + CX:HY = CY - Y
79Ø  IF IT = Ø THEN  HPLOT SX,HY:
     IT = 1
8ØØ  HPLOT SX,HY
81Ø  NEXT D
82Ø  NEXT T
825  GET D$
83Ø K = 9.8 * I * I / 2
84Ø  HPLOT CX,CY + K - 2 TO CX,CY
     + K + 2: HPLOT CX - 2,CY +
     K TO CX + 2,CY + K
845  FOR D = Ø TO 2 * 3.1415926 STEP
     Ø.3 / I
85Ø X = VØ * I *  COS (D):Y = VØ *
     I *  SIN (D) - K
86Ø SX = SC * X + CX:HY = CY - Y
87Ø  IF IT = Ø THEN  HPLOT SX,HY:
     IT = 1
88Ø  CALL 64477
89Ø : HPLOT SX,HY
892  NEXT D
893  PRINT "T=";T
896  INPUT "D=?1,?2,?3=";D
897  IF D = 2 THEN 9ØØ
898  IF D = 3 THEN  RUN
899  NEXT I
9ØØ  END
1ØØØ  INPUT "AØ=";AØ
1ØØ5  FOR I = 1 TO N
1Ø1Ø  PRINT "V(";I;")=";
1Ø11  INPUT V(I)
1Ø15 H(I) = V(I) * V(I) *  SIN (Ø
     .Ø1745 * AØ) *  SIN (Ø.Ø1745
     * AØ) / (2 * 9.8)
1Ø2Ø  IF V(I) * V(I) *  SIN (2 *
     Ø.Ø1745 * AØ) / 9.8 > 135 OR
     H(I) > 8Ø THEN 1Ø1Ø
1Ø25  NEXT I
1Ø3Ø  FOR I = 1 TO N - 1
1Ø4Ø  FOR J = I + 1 TO N
1Ø5Ø  IF H(I) > H(J) THEN H = H(I
     ):H(I) = H(J):H(J) = H:V = V
     (I):V(I) = V(J):V(J) = V
1Ø6Ø  NEXT J
1Ø7Ø  NEXT I
1Ø8Ø TØ = 2 * V(1) *  SIN (Ø.Ø174
     5 * AØ) / 9.8
1Ø85 TØ =  INT (TØ * 1ØØ) / 1ØØ
1Ø9Ø  FOR T = Ø TO TØ STEP TØ / 4
     Ø
11ØØ  FOR I = 1 TO N
111Ø X = V(I) *  COS (Ø.Ø1745 * A
     Ø) * T:Y = V(I) *  SIN (Ø.Ø1
     745 * AØ) * T - 9.8 * T * T /
     2

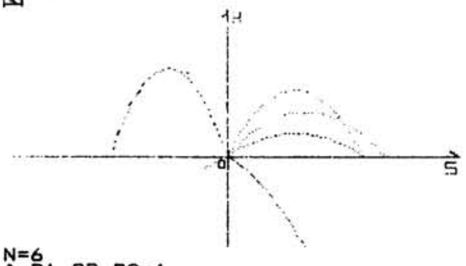
ULIST112Ø,

112Ø  HPLOT 13Ø + X,8Ø - Y
1125  CALL 64477
113Ø  NEXT I
114Ø  IF T = TØ / 4 OR T = TØ / 2
     OR T = 3 * TØ / 4 OR T = TØ
     THEN 116Ø
115Ø  NEXT T
116Ø  HCOLOR= 3
1165  GET C$
117Ø  HPLOT 13Ø,8Ø + 9.8 * T * T /
     2 TO 13Ø + X,8Ø - Y
```

```
1175  INPUT "E=?1,?2,?3=";E
1176  IF E = 2 THEN 1200
1177  IF E = 3 THEN  RUN
1180  GOTO 1150
1200  END
```

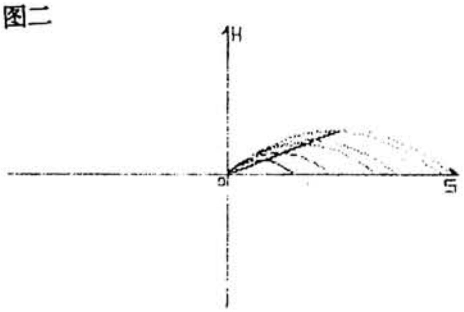
```
N=6
A=?1,?2,?3=1
A(1)=?30
V(1)=?30
H(1)=11.475   S(1)=79.522   T(1)=3.06
A(2)=?60
V(2)=?30
H(2)=34.43   S(2)=79.55   T(2)=5.301
A(3)=?45
V(3)=?30
H(3)=22.952   S(3)=91.836   T(3)=4.328
A(4)=?-30
V(4)=?25
H(4)=7.969   S(4)=55.224   T(4)=2.55
A(5)=?205
V(5)=?32
H(5)=9.304   S(5)=79.952   T(5)=2.755
A(6)=?110
V(6)=?32
H(6)=46.145   S(6)=67.106   T(6)=6.137
B=?1,?2,?3=1
```

图一



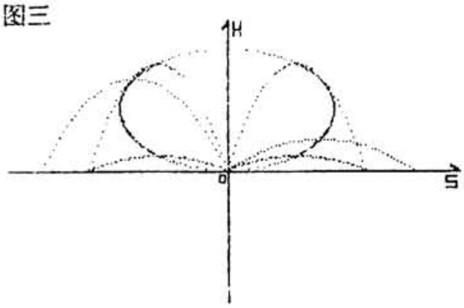

```
N=6
A=?1,?2,?3=1
A(1)=?35
V(1)=?20
H(1)=6.711   S(1)=38.35   T(1)=2.34
A(2)=?35
V(2)=?40
A(2)=?35
V(2)=?37
H(2)=22.971   S(2)=131.258   T(2)=4.33
A(3)=?35
V(3)=?25
H(3)=10.487   S(3)=59.924   T(3)=2.925
A(4)=?35
V(4)=?30
H(4)=15.101   S(4)=86.29   T(4)=3.511
A(5)=?35
V(5)=?32
H(5)=17.182   S(5)=98.18   T(5)=3.745
A(6)=?35
V(6)=?36
H(6)=21.746   S(6)=124.258   T(6)=4.213
B=?1,?2,?3=2
```

图二



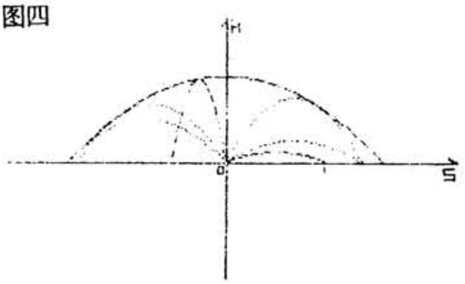

```
N=6
A=?1,?2,?3=1
A(1)=?20
V(1)=?35
H(1)=7.308   S(1)=80.334   T(1)=2.442
A(2)=?30
V(2)=?35
H(2)=15.619   S(2)=108.24   T(2)=3.57
A(3)=?70
V(3)=?35
H(3)=55.179   S(3)=80.392   T(3)=6.711
A(4)=?120
V(4)=?35
H(4)=46.896   S(4)=108.202   T(4)=6.187
A(5)=?110
V(5)=?35
H(5)=55.203   S(5)=80.278   T(5)=6.712
A(6)=?160
V(6)=?35
H(6)=7.332   S(6)=80.448   T(6)=2.446
B=?1,?2,?3=3
C=?1,?2,?3=1
```

图三




```
N=6
A=?1,?2,?3=1
A(1)=?20
V(1)=?30
H(1)=5.369   S(1)=59.022   T(1)=2.093
A(2)=?30
V(2)=?30
H(2)=11.475   S(2)=79.522   T(2)=3.06
A(3)=?60
V(3)=?30
H(3)=34.43   S(3)=79.55   T(3)=5.301
A(4)=?100
V(4)=?30
H(4)=44.538   S(4)=31.352   T(4)=6.029
A(5)=?135
V(5)=?30
H(5)=22.979   S(5)=91.836   T(5)=4.331
A(6)=?125
V(6)=?30
H(6)=30.829   S(6)=86.2720001   T(6)=5.01
B=?1,?2,?3=3
C=?1,?2,?3=2
C=?1,?2,?3=3
```

图四



软件赠送及I C芯片供应

我公司为了推动计算机事业的发展，免费向广大用户提供一批I B M P C及A P P L E Ⅱ的软件，清单如下：

一．I B M P C软件：

1．汉字WORDSTAR	1张
2．汉字SUPERCALC3	1张
3．汉字LOTUS 1—2—3	5张
4．汉字d BASE Ⅲ	1张
5．汉字d BASE Ⅱ 2．4 1	1张
6．汉字工程进度管理	1张
7．汉字工资管理	1张
8．汉字通用人事档案管理	1张
9．汉字固定资产管理	1张
1 0．COPY Ⅲ PC COPY Ⅱ PC	1张
1 1．汉字绘图软件	1张
1 2．人工智能GLISP	5张
1 3．汉字COBOL	1张
1 4．抓小偷	1张
1 5．武士道	1张
1 6．啤酒馆	1张
1 7．十项全能	1张
1 8．斗 蛇	1张

二．A P P L E Ⅱ软件：

1．英文d BASE Ⅱ	1张
2．CP／M2．2	1张
3．FORTRAN 8 0	1张
4．LOCKSMITH 4．1	1张
5．中国象棋	1张
6．抓小偷	1张

以上软件每张收工本费1 5元（含邮费）。需要者请汇款到：广州市海珠广场西侧2 9号，广州市计算机软件开发公司服务部，收款人：伍素娟。汇款时，请写明用户单位名称、地址，以及写明软件名称。复制软件者请不要把空盘寄来。

另外，我公司有大量进口I C芯电出售，清单为下：

8 0 3 9	1 8元／只
6 2 6 4	4 0元／只
6 1 1 6	1 5元／只
2 7 6 4	2 5元／只
Z 8 0 P I O	1 7元／只
Z 8 0 C T C	1 5元／只
Z 8 0 A	1 5元／只
微型开关	5元／只
数码管	3．5元

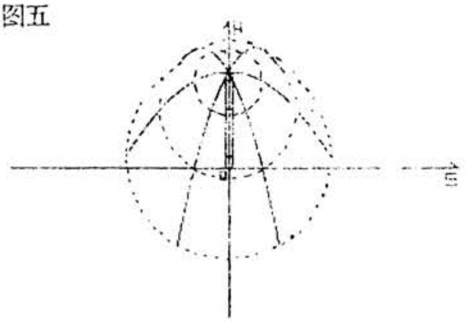
还有各种名牌软磁盘及J ANUS清洗盘出售，价格优惠，欢迎惠顾。

以上各货品若批发，价格另议。

广州市计算机软件开发公司服务部

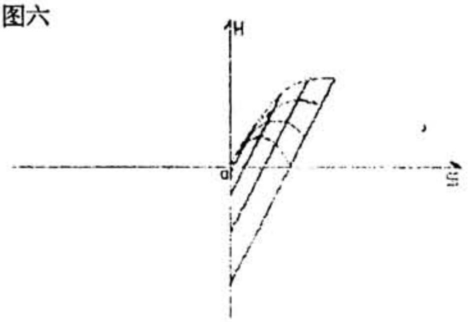
```
N=6
A=?1,?2,?3=2
VØ=18
T=1
D=?1,?2,?3=1
T=2
D=?1,?2,?3=1
T=3.01999998
D=?1,?2,?3=2
```

图五




```
N=4
A=?1,?2,?3=3
AØ=6Ø
V(1)=?2Ø
V(2)=?25
V(3)=?3Ø
V(4)=?35
E=?1,?2,?3=1
E=?1,?2,?3=1
E=?1,?2,?3=1
E=?1,?2,?3=2
```

图六



软磁盘驱动器维修经验一例

华中工学院计算中心
秦山秀　　闵艺华

此文提及的驱动器是用于ＩＢＭ－ＰＣ／ＸＴ微型计算机上的全高软磁盘驱动器，其型号为ＴＭ－１００－２Ａ。

根据我们对本单位有故障的软磁盘驱动器的维修情况发现，其故障基本上来自于“００道”微动开关。一般，人们如果发现“００道”微动开关有问题，采用的方法是重新更换一个“００道”微动开关。但有些情况下（如我们单位），该开关并不是真正地“坏死”了，而是由于该开关内部触点上已有灰尘，导致其接触不好。如果采取更换的方法，就显得有点浪费，并且会遇到以下问题：

a·该种“００道”微动开关在国内市场上很难买到。

b·“００道”微动开关从软磁盘驱动器上取下来后，再装一个上去，其位置很难装准确，需要认真，仔细地调整，否则，即使“００道”微动开关装上去了，该软盘驱动器仍然会出“００道”位置找不到的错误，不能读／写盘，且在调整“００道”微动开关位置时，比较费事。

我们在对本单位十几台驱动器属于“００道”微动开关故障的维修中，采用了一种非常简单，不用拆换“００道”微动开关的办法，使驱动器很快恢复了正常工作。下面介绍这种方法，以供大家参考。

一．“００道”微动开关的工作原理及检查方法

图四为“００道”微动开关内部与软盘驱动器控制器电路板上Ｐ１１的连接线路图。

在“００道”微动开关无故障，软盘驱动器工作正常的情况下，磁头小车处于“００道”位置时，磁头小车上的撞块应将“００道”微动开关触点压下，使图四中的“００道”位置与ＣＯＭ位置短接（即：Ｐ１１的1和4脚短接。而当磁头小车处于非“００道”位置时，磁头小车上的撞块应将“００道”微动开关释放（即：微动开关上的触点弹起，Ｐ１１的2和4脚短接），这样使驱动器能正确地找到００道位置。

根据“００道”微动开关的工作原理，可作以下检查，以判断是否是“００道”微动开关的问题。

1.将磁头小车移到驱动器上最后位置，使磁头小车上的撞块压下“００道”微动开关触点上，用万用表测量Ｐ１１的1和4脚之间的电阻和Ｐ１１的2和4脚之间的电阻，如果$R_{1,4}=0$，$R_{2,4}\neq 0$，则该状态下的“００道”微动开关没有故障，否则有故障。

2.将磁头小车向前移动，使磁头小车上的撞块离开“００道”微动开关上的触点，再用万用表测量，如果$R_{2,4}=0$，$R_{1,4}\neq 0$则该状态下的“００道”微动开关没有问题，否则有故障。

1987 IEEE
亚洲电子学会议即将在香港召开
——会议期间将举办《亚洲电子学周》展览

本刊讯　编委副主任委员陈兴业报道：1987年度IEEE　亚洲电子学会议将于9月1号到4号在香港召开。本次会议的论文选题范围涉及：

VLSI　设计与工艺；专用集成电路；表面装配工艺；电子仪器与遥感；自动测试；电能变换与控制；工业自动化与机器人；CAE，CAD与CAM；固态器件；微处理器与应用；人工智能；信号处理；计算机图象与模式识别；以及与电子工程和计算机科学有关的其它课题。

会议期间将同时举办《87年亚洲电子学周》展览会以及有关讲座。会议的技术程序委员会有13人组成：香港理工学院萧文治博士担任委员会的主席，我国林定基和沙宋（音译）二教授应邀担任该委员会的委员。

二．修理方法

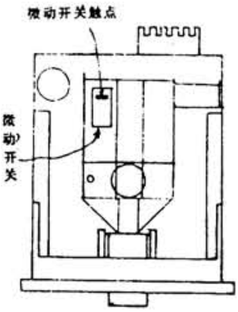
如果已查出是“00道”微动开关的故障，维修是非常简单的，方法如下：

用手（或起子）将“00道”微动开关的触点按下，然后放松，使之弹起，再按下，再放松，这样反复按动十几次，几十次后，再根据前面所谈的检查方法对“00道”微动开关进行检查。一般来讲，做如上工作后，“00道”微动开关是可以恢复正常工作的。

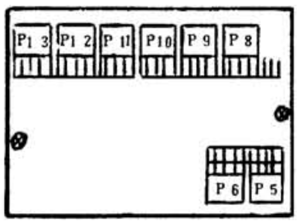
如果按动几十次后，仍不见效，就应考虑是否是“00道”微动开关彻底坏了。当然，在这种情况下，只有更换“00道”微动开关了。

值得提醒的是：用手或用起子按动“00道”微动开关上触点时，不能用力太大，否则会造成“00道”微动开关位置的移动。尽管微动开关已修复，磁头仍然找不到“00道”位置，驱动器依然不能正常的读／写操作。所以正确的操作是很必要的。

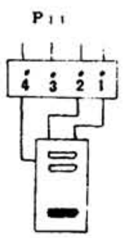
三．“00道”微动开关的位置及插图



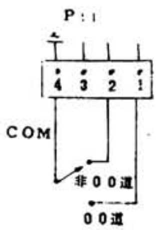
图（一）00道微动开关位置



图（二）驱动器控制板



图（三）P11与微动开关连接



图（四）　P11与微动开关内部结构

怎样直接在
中造

CC—DOS 汉字系统在国内各行各业广泛得到使用。根据汉字使用频度的统计，二级汉库已占使用汉字的99.9%以上。因此，CC—DOS 汉字系统基本上能满足大部分汉字信息处理系统的使用要求。但是也会碰到一些问题，对有些应用领域，如人事管理、医药管理以及专业词汇较强的应用系统，时常会遇到一些生僻字、异体字或不常用的科学符号等。为弥补这一不足，应用直接在汉字库中造字的方法，即使手中没有造字软件，同样可以根据自己的需要，造字或修改汉字。

CC—DOS 汉字库文件中共储存了6763个汉字和619个图形字符号。无论哪种版本的CC—DOS，都应有8000个汉字位置，占盘容量256K。因而剩有1000多个字模空间可以用来造字。这些汉字的点阵信息是按照GB2312—80区位码的位置顺序放置的，每个汉字是16×16点阵字模，因而每个字形点阵需要用256位二进制数，即由32个字节的数字来表示。在系统中每两个十六进制数表示一个字节，区位码中每个区有0—99位，从“汉字编码字符集”上看，每个区均有空位。譬如2区的0—16位都是空位（即32个字节均是“00”）。这些空位便可用来造新的汉字。当然，在其它有汉字信息的位置也可以对汉字修改。造字完成后，只要输入相应的区位码，就可以显示或打印出所造的新字来。

在汉字库中造字的具体步骤如下：

1，先在小方格纸上把所要造的字形以点阵方式描绘在16×16的方框内，如图一所示。例如，要造一个乜斜的“乜”（mie）字，在方格纸上绘出该字的点阵，涂黑的地方为二进制的“1”，未涂黑的地方为二进制的“0”，把八位一个字节换算为十六进制数以备使用。（如果是仿宋体字，应横杠画一排点，竖杠画两排点）。



图一

```
d0cc0
0CD3:0CC0  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
0CD3:0CD0  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
0CD3:0CE0  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
0CD3:0CF0  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
0CD3:0D00  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
0CD3:0D10  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
0CD3:0D20  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0CD3:0D30  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-
```

图二

```
0CD3:0CC0  18 00 1C 00 18 18 18 1C-18 FE 1B CC 1E 0C 38 4C   .........~.L..BL
0CD3:0CD0  D8 3C 18 18 18 10 18 00-18 02 18 06 0F FF 07 FE   X<..............
0CD3:0CE0  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
0CD3:0CF0  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
0CD3:0D00  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0CD3:0D10  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
0CD3:0D20  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
0CD3:0D30  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
```

图三

CC DOS汉字库
字或修改汉字

—上海机械进出口公司电脑部　李亦何

2，把CC—DOS的字库文件（CCLIB文件）拷贝到一张格式化过的空磁盘上，把这张软盘插入驱动器中，在西文状态下，用DEBUG命令把该字库调入内存。

A>DEBUG　CCLIB　［回车］

调入内存后，CS：100H开始对应汉字区位码的第一区第一位。下面　提供前几个区的地址，以便用户使用。

1 区，CS：0100H—0CA0
2 区，CS：0CC0—1860，
3 区，CS：1880—2420，
4 区，CS：2440—2FE0，
5 区，CS：3000—3BA0，
6 区，CS：3BC0—4760，

后面各区的地址均以十六进制的0 BC0 递增。

3，造字：例如在2 区的第1 位造图一所示的汉字，用显示命令D显示出如图二的信息。

—D0 CC0［回车］

在图二中可以看出，每一行有十六个字节，那么两行就可以放置一个汉字字形点阵。然后用E 命令把图一的点阵信息相应的3 2 个十六进制数顺序输入，输入完一个字节按一次空格键，直至输完后按回车。

—E0 CC0［回车］

××××：0CC0　00·

信息输入完毕后，再用D命令将显示图三所示的信息，如有错误，用E 命令再进行修改。

—D0 CC0［回车］

—W［回车］

Writing　37C40　bytes

—Q［回车］

如果没有错误，则用W命令把造好的字存盘，该汉字信息将永久保留在汉字库文件中。退出DEBUG，该汉字就造好了。

使用时，调入CC—DOS汉字系统，用区位码输入方式打入0201，便是刚造好的“乜”字。如果感到字体不够理想，仍可用同样方法进行修改。可想而知，在CS：0CE0 处造字，区位码则为0202，依此类推。只要造字者记住所造新字的地址对应的区位码，使用与原有汉字一样方便。这种方法虽然稍繁琐，但对于一般手中没有造字软件的用户，却是一种行之有效的方法。用户要是熟悉二——十六进制换算法，操作起来就会更得心应手。

十六进制		二进制	
18	00	00011000	00000000
1C	00	00011100	00000000
18	18	00011000	00011000
18	1C	00011000	00011100
18	FE	00011011	11111110
1B	CC	00011011	11001100
1E	0C	00011110	00001100
38	4C	00111000	01001100
D8	3C	11011000	00111100
18	18	00011000	00011000
18	10	00011000	00010000
18	00	00011000	00000000
18	04	00011000	00000100
18	06	00011000	00000110
0F	FF	00001111	11111111
07	FE	00000111	11111110



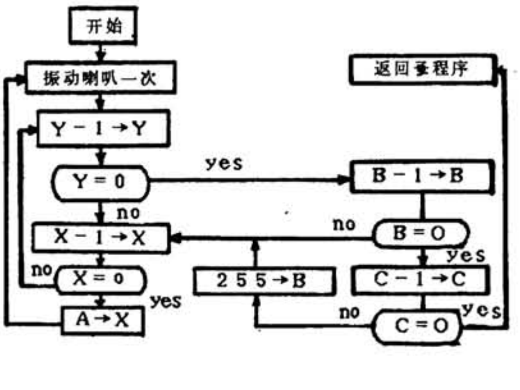
实现延长8位机喇叭发音时间的方法

福建省南平市科技情报研究所　　张国治

8位机的一个内存单元表示的值为0～255，因此，在8位机上编写音响程序时节拍稍长的音就不能实现。例|5－|假定控制一拍的时间值为160，则两拍之值为330，而控制单元表示不了这个数，如果将其分成两个一拍，则奏出的音乐别扭难听，效果极差。本文介绍的方法可使一个音要发多长就有多长，根本地解决了8位机的不足，从而使在音乐、游戏等音响效果上取得了良好的效果。并对一般具有BASIC基础者均可利用本结论编写出音乐程序，奏出悦耳动听的乐曲。以下所述在APPLE II上通过。

设单元A用以控制音阶的高低，单元B用以控制节拍的长短，X及Y为6502 CPU的两个8位寄存器，现用第三个单元C与B构成表示以255为基数的数，C为高位，B为低位。则一拍值160就是C为0，B为160。两拍之值330就是C为1，B为75。为了方便对单元C＝0的判断，具体算法是：

当节拍控制值>255时，C＝1＋INT（控制值／255），B＝控制值－（C－1）×255，否则C＝1，B＝控制值。



发音子程序的框图

显然CB可表示的值为0～255×254＋255之间，足以实现乐曲中尽管拉得很长的音节。

上面图是8位机通用的发音子程序框图。依照框图，不同的8位机可用各自的CPU指令写出程序。APPLE II的程序清单见本文末。

设A、B、C分别表示内存单元$300，$301及$302。

```
303  LDA  C030
306  DEY
307  BEQ  312
309  DEX
30A  BNE  306
30C  LDX  300
30F  JMP  303
312  DEC  301
315  BNE  309
317  DEC  302
31A  BNE  31D
31C  RTS
31D  LDA  #FF
31F  STA  301
322  JMP  309
```

该子程序用十进制数写出的34个数据是后面实例程序中10～14语句的数据。

一个实例　　|5. 6|3 2|1－|1 －|。

	|5·	6|	3	2|	1－|1－|
音阶控制值：	128	114	152	172	192
节拍控制值：	255	70	160	160	660
因此 A：	128	114	152	172	192
B：	255	70	160	160	150
C：	1	1	1	1	3

每一音符的A、B、C值均可由自动换算程序来实现（略）。相应音阶、节拍的控制参考值附后。实现演奏程序如下，其中10～14为发音子程序用十进制数写出的34个数据，20～26建立发音子程序，68～80演奏程序段。

IBM－PC／XT常见故障与维修（二）

华南师范大学　张毅忠、郑存陆、罗家耀

4.延时器TD 1 输出脚无波形

图7 示出系统板上主要的地址解码电路。图的上方U_{45}（74LS138）是用于I／O地址译码，它产生DMA控制器（8237A－5）选片信号$\overline{DMACS}$，中断控制器（8259A）选片信号$\overline{INTRCS}$，定时／计数器（8259－5）选片信号$\overline{T/CCS}$，和可编程的I／O（8255A－5）选片信号$\overline{PPICS}$。图的中间部分电路主要包括U_{44}，U_{42}，TD1，U_{56}和U_{55}等IC片，构成RAM地址译码，产生RAM的行选$\overline{RAS0}$～$\overline{RAS3}$和列选$\overline{CAS0}$～$\overline{CAS3}$信号。图的下方U_{43}等

```
ULIST

10  DATA  173,48,192,136,240,9,20
     2,208,250,174,0,3
12  DATA  76,3,3,206,1,3,208,242,
     206,2,3,208,1,96
14  DATA  169,255,141,1,3,76,9,3
20  FOR I = 1 TO 34
22  READ A
24  POKE 770 + I,A
26  NEXT
35  FOR·I = 1 TO 1000
40  READ A(I,0),A(I,1),A(I,2)
42  IF A(I,0) = 0 THEN 50
45  NEXT
50 N = I - 1
60  FOR I = 1 TO 3
68  FOR J = 1 TO N
75  POKE 768,A(J,0): POKE 769,A(J
     ,1): POKE 770,A(J,2)
80  CALL 771: NEXT J
85  NEXT I
90  END
95  DATA 128,255,1,114,70,1,152,1
     60,1,172,160,1,192,150,3
1000  DATA  0,0,0
```

附：音阶、节拍控制参考值：

音阶	1	2	3	4	5	6	7	i	o
控制值	192	172	152	144	128	114	102	96	1

未列出音阶的控制值按1～7，高八度除2，低八度乘2。

节拍	1／4	1／2	3／4	1	1.5	2	4
控制值	30	70	110	160	255	330	660

休子符“0”的控制值1拍为80，半拍为38。

ＩＣ片主要用于ＲＯＭ地址译码，产生$\overline{CS6}$、$\overline{CS7}$信号。

根据电路图可以得出：$RAS=\overline{XMEMW}+\overline{XMEMR}$的逻辑关系，即在存贮器读操作时（$\overline{XMEMR}=1$）或者在存贮器写操作时（$\overline{XMEMW}=1$），均产生行选信号RAS（＝1）。RAS输入延时器TD1，经60ns延时后，产生地址选通信号ADDRSEL；100ns延时后，输出一个供产生列选$\overline{CAS}$的信号。$\overline{CAS}$须在地址允许$\overline{AEN}=0$及$\overline{MEMR}=0$，$\overline{MWTC}=0$时，方能产生。$\overline{CAS}$控制着U42IC片的启动，而ADDRSEL信号控制U39和U40（见图5）的传送方向。

显然，$\overline{CAS}$和ADDRSEL信号是否正常，直接影响到随机存贮器（RAM）能否正常工作，换句话说，TD1损坏就使得RAM无法进行读写操作。在自检基本16KBRAM时，就出错，机器停运，屏幕无显示。当用示波器测得TD1的脚1（输入信号）波形正常，而输出脚10，8无波形时，就可以断定延时器TD1损坏，更换TD1后再试。图6示出TD1脚1的正常波形，而脚10，8的波形与之相同，仅相位稍有滞后而已。



图6　TD1延时器脚1正常波形

5．RAM的地址多路器U_{39}、U_{40}无输出或数据缓冲器U_9输入／出不正常

图9示出系统板中的RAM地址及数据线与总线连接情况。在系统中采用64K×1bit的4164芯片时，多路器U84没有设置（U84是用在256K×1的RAM芯片系统）。16根地址线依靠U39和U40（均为74LS158）多路器分时传送行、列各8根地址信息。当U39及U40的脚1为低电平时，输出接通输入端的A路；反之，接通输入端的B路。U39，U40的脚1是与由RAS延时60ns后产生的ADDRSEL信号相连，所以波形如同TD1的脚10。

数据缓冲器U_9（74LS245）是由信号$\overline{RAMADDRSEL}$来选通，其数据传送方向是由信号$\overline{XMEMR}$来决定：$\overline{XMEMR}=0$，数据从RAM读出；反之数据写入RAM。

在数据写入RAM操作时，由U_{20}（S280）IC片自动产生偶校验位值（MDPIN），将其写入RAM的奇偶校验位；在读出RAM的数据操作时，8位数据和奇偶校验位之值一同送入U_{20}进行校验。如果发现有错，U_{20}的脚6为0，该信号送到触发器U_{86}，然后产生PCK和$\overline{PCK}$两个互补的信号，它们再经过有关的逻辑电路作用，引到CPU8088的非屏蔽中断（NMI）脚，使系统产生NMI中断。

开机后系统自动对基本16KRAM进行检测，如果多路器U_{39}或U_{40}损坏，或者数据缓冲器U_9损坏，乃至只损坏一根数据线引脚，都会引起检测错误。这种错误，不是发生于RAM芯片，而是U_{39}或U_{40}或U_9IC芯片。按系统开机自检程序的顺序可知基本16KRAM检测出错，系统将停运，屏幕无显示，也无喇叭响声。笔者曾遇到以下两种情况，并断定其U_9，U_{39}损坏：

①U_9损坏。开机后系统不启动无显示，用示波器测试TD1输入／出正常。U_{39}和U_{40}的输入／出正常。U_{42}无选片信号输出，全为高电平。U56与U_{42}情况相同。U_9的脚1、19有信号正常如图8所示。U_9的数据线引脚除一对线（脚9、11）有信号外，其余均为高电平。

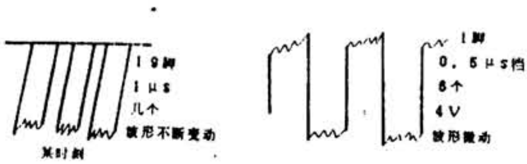


图8　U_9的脚1，19正常波形

②U_{39}损坏。开机后系统不启动无显示，用示波器测得：TD1输入／出正常；U_{39}的输出脚4，7，9，12无波形，而输入脚信号正常；U_{40}的输入／出均正常。

③U_{40}损坏。开机后系统不启动无显示，用示波器测试多数波形正常，U_{40}的输出脚4，7，9亦正常，但输出脚12的波形暗弱且闪动。最后查出除U_{40}损坏外还有一只RAM芯片损坏。

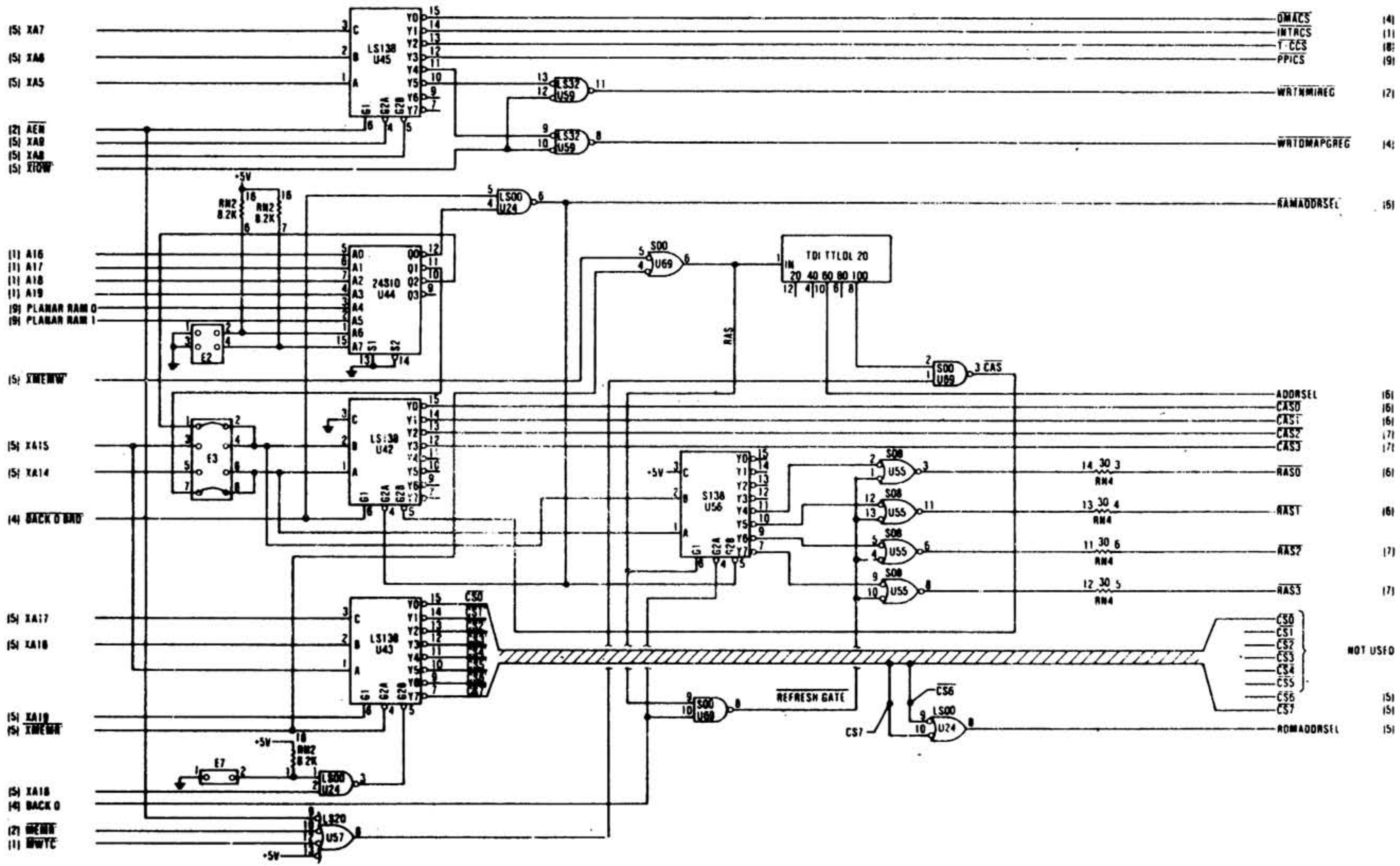

LS138
U45
24S10
U44
LS138
U42
LS138
U43
S138
U56
TDI TTLDL 20
DMACS
INTRCS
T CS
PPICS
WRTNMIREG
WRTDMAPGREG
RAMADDRSEL
ADDRSEL
CAS0
CAS1
CAS2
CAS3
RAS0
RAS1
RAS2
RAS3
CS0
CS1
CS2
CS3
CS4
CS5
CS6
CS7
NOT USED
ROMADDRSEL
REFRESH GATE
RAS
XA7
XA6
XA5
AEN
XA9
XA8
A16
A17
A18
A19
PLANAR RAM 0
PLANAR RAM 1
XA15
XA14
BACK O BRD
XA17
XA16
XA18
BACK O
+5V
RN2
8.2K
RN4
E2
E3
E7

图7．系统板接线图（3）

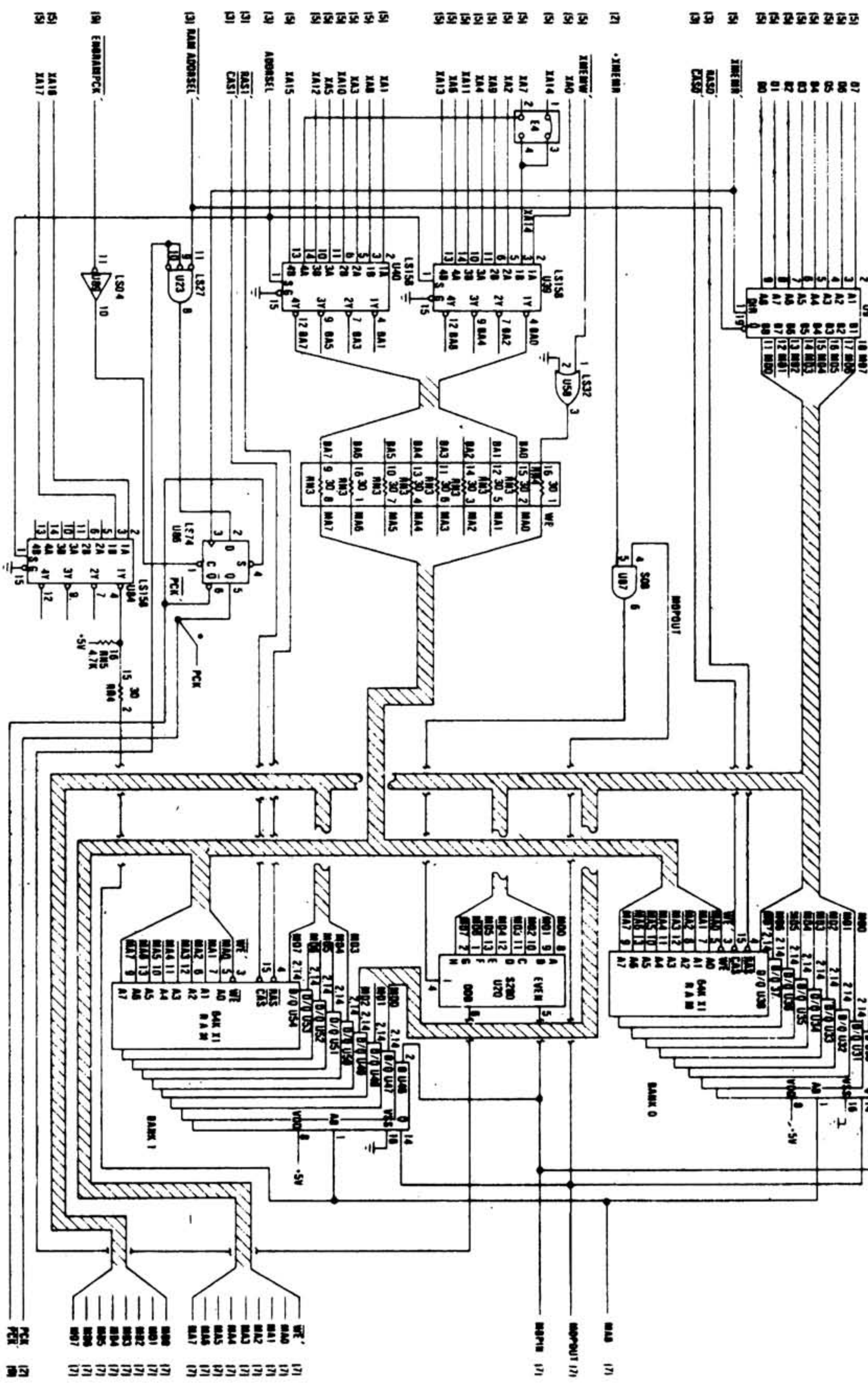


图9　系统板接线图（6）

迷宫问题新解

西安交通大学计算机研究室　龙安川

前　言

迷宫可以看成是由一些障碍物（如图1 的黑色单元）和通道（如图1 的白色单元）组成的矩形阵列，它指定了一个入口和一个出口，要求从入口开始，每次只能从一个白色单元移到另一个相邻的（上下左右相邻）白色单元，直到找到一条通往出口的路径为止。

通常用计算机来解决这个问题时，用的是一步步试探的方法，即每走一步，检查其相邻单元是否可通行以及以前是否走过，如果可通行且以前没有走过，则继续走，否则退回一步重新试探下一个位置，如此重复进行直至出口。如果退回到入口处了，则表明无路可走。

如果我们只要求能寻找到一条通往出口的路径就可以了，那么上述算法基本上是满足要求的，但如果存在多条路径时，我们不仅要求能寻找到一条通路，而且还必须是所有通路中最短的一条，若仍用试探法就不合适了。当然我们也可以用其他方法，如用邻接矩阵来计算最短路径，但这种方法需要很大的存贮空间，如本例的１０×１０方阵，光邻接矩阵就需要１００×１００的存贮空间，如果迷宫为１００×１００方阵，邻接矩阵就需要１００００×１００００的存贮空间，这在一般计算机上简直是不可想象的。下面我要介绍一种方法，借助这种方法我们不仅能计算出通往出口的最短路径的长度，而且还能找到一条通往出口的最短路径，所需的存贮空间很少，在一定程度上却比试探法速度要快。

一）算法介绍

我们知道波是通过波源的振动带动其相邻的质点振动，然后又以这些质点为新的波源带动各自相邻的外围质点振动，从而由近及远地向各个方向传播出去。如果把这种现象用于迷宫中会是怎样的呢？这里我们用自然数列（即0，1，2，……）代表波的波峰序列来模拟波的传播过程。（参见图2）

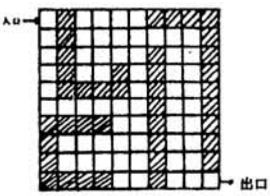


图1

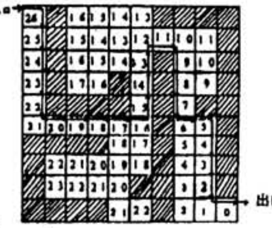


图2

首先以出口单元作为第一波峰ϕ_1的波源，出口单元本身则为零波峰ϕ。零波峰单元的相邻单元未被占据的单元都是第一波峰，将它们打上数列的第一元素记号，即数字1。形成ϕ_1的单元都是第二波峰ϕ_2的波源，所有和波峰ϕ_1相邻的未被占据的单元，都得到了第二波峰的标志，即数字2。波绕过所有障碍物，从零波峰这个波源向四面八方扩散，波的构成过程一直要持续到到达入口单元为止，也就是入口单元也依次轮到成为波峰的时候。如果到达入口单元，那么出口和入口之间便有通路存在。否则就无路可走。数列可以保证任两个波峰都不能打上相同的标号，算法工作的第二步骤就是利用这个道理。

如果有通路存在，那么在第二阶段就是确定其总和为构成所求通路的单元序列。建立通路的实质就是按照一定规则，从入口单元$\in \phi_K$向同它邻接的单元$j \in \phi_{K-1}$等等，顺序地沿着邻接的波峰单元延伸到零波峰单元（即出口），它的走向和波的传播方向相反，按波峰脚标的“递减”数字，即$K \rightarrow K-1 \cdots \rightarrow 3 \rightarrow 2 \rightarrow 1$等等。在建立通路时，从第K波峰的单元到第K－1 波峰的单元出现非单一的出口。为了消除这种非单一性，可以为每一单元的相邻单元规定优先分析顺序。如东、南、西、北，遵循所取的相邻单元优先分析顺序和波峰脚标“递减”序列，得到了一条连线，其线路如图2 所示的箭头所指。如果改变相邻

单元优先分析顺序，可以得到其它路线。路径的长度即为入口单元波峰的序号。

数列也可以用周期数字，如1，2，3，1，2，3，1，…等等，这样可以减少每个单元所需存贮容量（如我们用数组表示迷宫的话，则每个单元只需2 b i t 即可以表示了），但在计算路径长度时增加了计算转折点的次数和时间，所以在满足所需存贮容量的基础上，数字周期值越大越好。

上述算法因为其与波的传播原理相同，所以通常称之为波动算法。

二）编程技巧

算法中需要把最外围的波峰位置记住，以便能继续扩散。为此需要定义两个堆栈，一个堆栈用于存贮波源位置，另一个堆栈用于存贮新的波源位置。为了节省存贮空间，这里我们用到了共享堆栈的概念，即用一个数组的两边分别作为两个堆栈的栈底，两堆栈都向中间伸展。（如图3 所示）

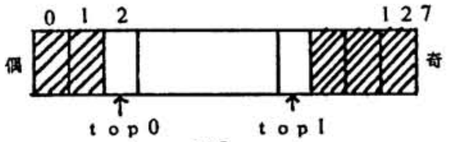


图3

程序的主要设计思想是：以ϕ_0为当前单元，把ϕ_i的所有单元入栈，然后逐个出栈作为当前单元，把其相邻的未被占据单元入栈，这样当ϕ_1的单元全部出栈后，ϕ_2的全部单元就已经全部压入到另一个栈中，同样把ϕ_2的单元逐个出栈作为当前单元，把其相邻的未被占据单元入栈，当ϕ_2的全部单元都出栈后，ϕ_3的全部单元就全部压入第一个栈中了。两个堆栈交替使用。若在第L 步要压入栈的单元为入口单元时就打印出最短路径长度和最短路径。当需要一个单元出栈，而这时如果两个堆栈全为空时，表明从出口扩散到了其所能播及到的每一个角落，但还是没有到达入口，也就是说，这两点之间没有通路。

下面就程序中的一些细节问题说明一下：

迷宫是用一个二维数组MAZE来存放的，其中通道用0 来表示，障碍物用7 来表示，入口和出口位置分别放在（E（0，0），E（0，1））和（E（1，0），E（1，1））中。注意障碍物之所以用7 来表示不是随便定义的，它是建立在我们假定每个迷宫单元为3 b i t 存贮空间的基础上的，因为3 b i t 可以表示8 种状态，即0 ～7，其中最小和最大值分别为0，7 用于表示通道和障碍物，剩下还有1，2，3，4，5，6 未用，按照前面所说的数字周期值应越大越好的原则我们把数字波数列定为1，2，3，4，5，6，1，2，3，4，5，6，1，…。如果我们假定每个迷宫单元为4 b i t 存贮空间，那么我们就可以用最小和最大值0，15 来分别表示通道和障碍物，剩下作为数列项。这一点在用机器语言实现上述算法时是很有用的。

为了避免经常检测是否出界，我们把迷宫四周都用障碍物围了起来。

具体工作过程请见程序中注释。

三）波动算法应用简述

波动算法应用面很广，特别是在大平面多节点图中，由于计算机存贮容量的限制，不可能用邻接矩阵来计算最短路径问题时，常用到此算法。如印刷电路板CAD中需要寻找任意两点间的最短路径的问题，在这里用的就是波动算法。波动算法在游戏程序中用的更是广泛，如我们所熟悉的《警察抓小偷》《PACKMAN》等游戏都用到了此算法或其导出算法（如射线算法等）。由于篇幅所限这里我就不一一列举了，以后如果有机会我将就用此算法编制大型游戏程序作进一步讨论和举例。有兴趣的读者不妨也试试看。

```
10 DIM MAZE(63,63),STACK(127,1),E(1,1)
20 READ E(0,0),E(0,1),E(1,0),E(1,1)        '读入入口和出口位置
30 READ M,N                                '读入迷宫大小
40 '读入迷宫
50 FOR J=1 TO N
60 FOR I=1 TO N
70 READ MAZE(I,J)
80 NEXT I
90 NEXT J
100 '迷宫四周设置成障碍物
110 FOR I=0 TO N+1
120 MAZE(I,0)=7:MAZE(I,N+1)=7
130 NEXT I
140 FOR J=0 TO N+1
150 MAZE(0,J)=7:MAZE(N+1,J)=7
160 NEXT J
170 TOP0=0:TOP1=127                        '初始化堆栈指针
180 L=-1:K=0                               '置数列初值
190 MAZE(E(1,0),E(1,1))=7
200 '出口单元填入7，表明出口单元已被占据，本应该填入0，但0与
通道数字冲突
210 X=E(1,0):Y=E(1,1):GOSUB 300            '以出口位置作为零波峰入
栈
220 GOSUB 490                              '波源单元出栈，作为当前
单元
```

```
230 '检测当前单元的相邻单元是否已被占据，没有则填入当前波峰值并入栈
240 IF MAZE(CX+1,CY)<>0 THEN 270
250 MAZE(CX+1,CY)=K
260 X=CX+1:Y=CY:GOSUB 380
270 IF MAZE(CX,CY+1)<>0 THEN 300
280 MAZE(CX,CY+1)=K
290 X=CX:Y=CY+1:GOSUB 380
300 IF MAZE(CX-1,CY)<>0 THEN 330
310 MAZE(CX-1,CY)=K
320 X=CX-1:Y=CY:GOSUB 380
330 IF MAZE(CX,CY-1)<>0 THEN 360
340 MAZE(CX,CY-1)=K
350 X=CX:Y=CY-1:GOSUB 380
360 GOTO 220
370 '----------入栈子程序--------------
380 IF (X=E(0,0)) AND (Y=E(0,1)) THEN RETURN 690
390 '检测波是否已扩散到了入口，是则输出计算结果
400 IF (K MOD 2)=0 THEN 450
410 '如果当前波峰值为偶数，则压入左侧堆栈，否则压入右侧堆栈
420 STACK(TOP1,0)=X:STACK(TOP1,1)=Y
430 TOP1=TOP1-1
440 RETURN
450 STACK(TOP0,0)=X:STACK(TOP0,1)=Y
460 TOP0=TOP0+1
470 RETURN
480 '----------出栈子程序--------------
490 IF (K MOD 2)=0 THEN 580
500 '如果当前波峰值为偶数，则其波源脚标必为奇数，所以应从右侧堆栈取数
510 IF TOP0=0 THEN 550          '检测左侧堆栈是否为空，为空则从右侧堆栈取数
520 TOP0=TOP0-1
530 CX=STACK(TOP0,0):CY=STACK(TOP0,1)
540 RETURN
550 IF TOP1=127 THEN RETURN 660          '如果两个堆栈全为空，则表明无路可走
560 L=L+1:K=1+(L MOD 6)          '计算下一波峰值
570 GOTO 590          '从右侧堆栈取数
580 IF TOP1=127 THEN 620          '检测右侧堆栈是否为空，为空则从左侧堆栈取数
590 TOP1=TOP1+1
600 CX=STACK(TOP1,0):CY=STACK(TOP1,1)
610 RETURN
620 IF TOP0=0 THEN RETURN 660          '如果两个堆栈全为空，则表明无路可走
630 L=L+1:K=1+(L MOD 6)          '计算下一波峰值
640 GOTO 520          '从左侧堆栈取数
650 '
660 PRINT "没有路径!"
670 END
680 '----------输出计算结果--------------
690 MAZE(E(1,0),E(1,1))=0          '出口单元恢复成0
700 '输出波的传播图
710 LPRINT "波的传播图为:"
720 FOR J=1 TO N
730 FOR I=1 TO M
740 LPRINT MAZE(I,J);" ";
750 NEXT I
760 LPRINT
770 NEXT J
780 LPRINT "最短路径长度 =";L+1          '输出最短路径长度
790 '按照先东南后西北的优先分析顺序和脚标递减的顺序寻找出口路径，并打印路径序列
800 CX=E(0,0):CY=E(0,1)
810 LPRINT "最短路径为:"
820 LPRINT "(";CX;",";CY;")";"--";
830 L=L-1
840 IF L=-1 THEN 930          '检测是否已到达出口，到达了就结束
850 K=1+(L MOD 6)
860 IF MAZE(CX+1,CY)=K THEN 920
870 IF MAZE(CX,CY+1)=K THEN 910
880 IF MAZE(CX-1,CY)=K THEN 900
890 CY=CY-1:GOTO 820
900 CX=CX-1:GOTO 820
910 CY=CY+1:GOTO 820
920 CX=CX+1:GOTO 820
930 LPRINT "(";E(1,0);",";E(1,1);")"
940 END
1000 '入口和出口位置；迷宫大小
1010 DATA 1,1,10,10,10,10
1020 '迷宫阵列
1030 DATA 0,7,0,0,0,0,7,7,7,7
1040 DATA 0,7,0,0,0,0,0,0,0,7
1050 DATA 0,7,0,0,0,0,7,0,0,7
1060 DATA 0,7,0,0,7,0,7,0,0,7
1070 DATA 0,7,7,7,7,0,7,0,7,7
1080 DATA 0,0,0,0,0,0,7,0,0,7
1090 DATA 7,7,7,7,0,0,7,0,0,7
1100 DATA 7,0,0,0,0,0,7,0,0,7
1110 DATA 7,0,0,0,0,7,7,0,0,7
1120 DATA 7,7,7,7,0,0,7,0,0,0
```

波的传播图为:

```
2  7  4  3  2  1  7  7  7  7
1  7  3  2  1  6  5  4  5  7
6  7  4  3  2  1  7  3  4  7
5  7  5  4  7  2  7  2  3  7
4  7  7  7  7  3  7  1  7  7
3  2  1  6  5  4  7  6  5  7
7  7  7  7  6  5  7  5  4  7
7  4  3  2  1  6  7  4  3  7
7  5  4  3  2  7  7  3  2  7
7  7  7  7  3  4  7  2  1  0
```

最短路径长度 = 26

最短路径为:

```
(1,1)--(1,2)--(1,3)--(1,4)--(1,5)--(1,6)--(2,6)--(3,6)--(4,6)--(5,6)--(5,5)--(6,5)--(6,4)--(6,3)--(6,2)--(7,2)--(8,2)--(8,3)--(8,4)--(8,5)--(8,6)--(9,6)--(9,7)--(9,8)--(9,9)--(9,10)--(10,10)
```

APPLE Ⅱ 高分辨率造型与色彩分析

浙江水利水电专科学校 陈中藩

在APPLE－Ⅱ系列微型机显示器上可以利用APPLE SOFT程序设计语言作图并可设定彩色，但往往得不到理想的效果，有时竟会出现不连续的彩色线条或色彩混乱的画面。目前有一些软件产品可利用高分辨率作图形式，在屏幕上显示一些特殊的符号（包括汉字），但座标位置也较死板，更无法确定其色彩。

本文对高分辨率造型和彩色设定中的一些特殊规律进行分析。了解和掌握这些规律后，就可以正确应用APPLE－Ⅱ的作图和色彩功能。

一．内存分配规律

在内存大于24K的APPLE－Ⅱ中，开辟有两个高分辨率图形区，分别称第一页和第二页，它们的地址分别是：8192（2000H）～16383（3FFFH）和16384（4000H）～24575（5FFFH）。各占8K内存。 所谓分辨率是对显示器上图形的象素（作图的最小光点）疏密而言，APPLE－Ⅱ上的高分辨率作图是指显示器屏幕上以一个基本光点作为图形象素，整个屏幕由水平方向280个象素；垂直方向192个象素组成图形的坐标系统，称之为280列、192行，共有280＊192＝53760个象素。

每行的280列，仍以40个内存单元（称为列单元）的低七位（第0位至第六位）与其一一对应，而列单元的最高位（第七位）用来设定图形的色彩。这样整个屏幕占用192＊40＝7680个内存单元（7.5K）。所以图形区每页的8K内存单元中有0.5K（512个单元）是弃去不用的。

每页的7.5K内存单元地址不是按顺序与图形坐标的行列对应的，其规律较为复杂。

首先，把垂直方向的192行分为三区（第0区、第1区、第2区），而每区分为八段（第0段～第7段），每段分为八行（第0行～第7行）。第0区第0段第0行，即图形坐标第0行的行首地址是8192（2000H）或16384（4000H），接着是第1区第0段第0行，即图形坐标的第64行，行首地址是8192＋40＝8232（2028H）或16384＋40＝16424（4028H），再是第2区第0段第0行，即图形坐标的第128行。行首地址是8232＋40＝8272（2050H）或16424＋40＝16464（4050H）每行40个列单元，共用去120个单元，然后弃去8个单元，返回到第0区第1段第0行，即图形坐标的第8行，行首地址是8192＋128＝8320（2080H）或16384＋128＝16512（4080H），接着是第1区第1段第0行，即图形坐标的第72行，……第2区第1段第0行，即图形坐标的第136行……弃去8个单元，又返回到第0区第2段第0行……最后是第0区第7段第7行到第1区第7段第7行到第2区第7段第7行，即图形坐标的最后一行第191行。每次返回第0区时都弃去8个单元，（其返回64次，弃去512个单元）。

由上述规律可知：每区对应行之间相差40个单元（28H）；区内每段对应行之间相差128个单

邮购消息

（接封二）

CAC—3学生电脑

CPU、Z80A、ROM、16KB（其中BASIC语言8KB、RAM16KB，可扩展至32KB）。42键，一键多用，可接监视器，并可直接与家用电视机相接，可配接磁带机、扩展器、打印机、手控游戏棒。

可用BASIC语言和Z80汇编语言。

单价295元／台，邮费包装费10元／台，批量从优。

邮购部地址：请见封二

型号	单价
集成块	
74LS32	1.5
74LS74	2
74LS138	3
74LS244	3.5
74LS367A	2.5
NE555	2
CD4001UBE	2.2
CD4002BE	2
CD4008BE	4.5
CD40107BE	4.5
学生电脑专用	
C4005	38

元(8 0 H) ；段内每行之间相差1 0 2 4 单元(4 0 0 H) 。

于是，已知图形坐标系统中的行数(Y) ，求该行首地址的公式如下：

行首址=页首址+4 0 *I NT(Y／6 4) +1 2 8 *I NT((Y—I NT(Y／6 4) *6 4) ／8) +1 0 2 4 *(Y—I NT(Y／8) *8)

以第一页图形区为例，第1 0 0 行的首地址是：

8 1 9 2 +4 0 *I NT(1 0 0 ／6 4) +1 2 8 *I NT((1 0 0 —I NT(1 0 0 ／6 4) *6 4) *1 8) +1 0 2 4 *(1 0 0 —I NT(1 0 0 ／8) *8) =1 2 8 4 0 (3 2 2 8 H)

图形象素所在的列单元，可以由行首址和图形坐标的列数(X) 求得，公式如下：

列单元地址=行首址+I NT(X／7)

仍以第一页图形区为例，第1 0 0 行，1 0 0 列的象素所在列单元地址为：

1 2 8 4 0 +I NT(1 0 0 ／7) =1 2 8 5 4 (3 2 3 6 H)

至于这个象素在列单元的第几位(第0 位起算) ，可以由下式求得：

位数=X—I NT(X／7) *7

即除7 取整，上例中的象素在列单元中的位数就是：

1 0 0 —I NT(1 0 0 ／7) *7 =2

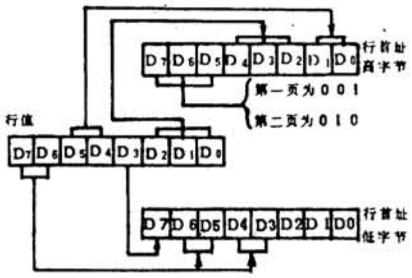
即在第1 2 8 5 4 (3 2 3 6 H) 单元的第2 位。

要注意：显示器接口电路是从低位到高位(即单元中二进制码的右面到左面，串行送数的，而显示器电子束是从屏幕的左面到右面扫描的。位值=0 ，不亮；位值=1 ，发亮。上例中第1 0 0 行、1 0 0 列象素如果是个亮点，那么第1 2 8 5 4 (3 2 3 6 H) 单元的低七位二进制值应为：0 0 0 0 1 0 0 。

上面算式是以十进制数用程序设计表达式形式描述的，便于利用A P P L E S O F T语言中的P O K E语句直接向指定单元置数。

如果在监控程序状态下，要求出行首地址的十六进制值，可用下列方法。

首先把行数化为一个字节的二进制值，然后把第0 ～第2 位三个数移至行首地址的高字节第2 ～第4 位(即乘上1 0 2 4 =2^{10}) ；把第4 、5 位移至高字节的第0 、1 位；把第3 位移至行首地址的低字节第7 位(即乘上1 2 8 =2^{7}) ；最后把第6 、7 位同时移至低字节的第5 、6 位和第3 、4 位，(即乘上4 0 =2^{5} +2^{3}) 。移位情况如图一。



图一 行首地址

在高字节的第5 ～7 位，由图形页面决定，第1 页为0 0 1 ，第2 页为0 1 0 。再根据列数除7 ，取整后，转换为十六进制值，加上行首地址，就得到列单元地址了。

二. 色彩的设定

彩色显示器上某个发亮的象素色彩与电子束击中的位置有关，在A P P L E －II微型机的显示器接口电路中用每个列单元的最高位来控制色彩的相位(色相) ，有下列规律：

列单元第七位值= { 0 { 偶数列亮，则为紫色。奇数列亮，则为绿色。 0 { 偶数列亮，则为蓝色。奇数列亮，则为桔色。

如果同一行有两个相邻的亮点，也就是奇偶两个连续的象素列都亮，则显示白色(尽管两个象素不在同一个列单元) 。

由于最高位同时控制着该单元的低七位，所以同一列单元的各象素不可能有蓝色与绿色，蓝色与紫色，或者橙色与紫色，橙色与绿色同时存在的现象。因此，在同一行绘制一条单色彩(除黑白两色外) 的水平线条，其象素最多仅有2 8 0 ／2 =1 4 0 个(仅有偶数列或者仅有奇数列) 。从而使彩色图形的分辨率下降了一半。但由于光点的衍射作用，视觉仍为一条“连续”的单色线条。

例如，把第一页高分辨率图形区的首行首列单元(2 0 0 0 H) 置成不同二进制数就可以显示不同色彩的一小段水平线，如表一所示：

要注意，当一行上的两个相邻的列单元置成同样的数值时，在彩色屏幕上显示的不是相同色彩的线段，而是两段不同色彩的线段。例如第0 行、0 列单元(

表一

色彩	二进制值								十六进制值	象素
黑色	0	0	0	0	0	0	0	0	0 0	7
紫色	0	1	0	1	0	1	0	1	5 5	4
绿色	0	0	1	0	1	0	1	0	2 A	4
白色	0	1	1	1	1	1	1	1	7 F	7
黑色	1	0	0	0	0	0	0	0	8 0	7
蓝色	1	1	0	1	0	1	0	1	D 5	4
桔色	1	0	1	0	1	0	1	0	A A	4
白色	1	1	1	1	1	1	1	1	F F	7

2 0 0 0 H）置成0 1 0 1 0 1 0 1（5 5 H），把该行第一列单元（2 0 0 1 H）也置成0 1 0 1 0 1 0 1（5 5 H），结果第一页第0 行从0 ～5 列是一段紫色水平线，第6 、7 两列为一小段白色水平线，而第8 ～1 3 列是一段绿色水平线，为什么会有这样的结果呢?

把这两个相邻列单元的低七位数值和位数及图形列数对应列出，如表二所示：

表二

地址	2000H							2001H						
二进制值	1	0	1	0	1	0	1	1	0	1	0	1	0	1
位数	0	1	2	3	4	5	6	0	1	2	3	4	5	6
列数	0	1	2	3	4	5	6	7	8	9	10	11	12	13

显然，第0 列单元（5 5 H）是把图形坐标的偶数列置成1 ，而且最高位为0 ，所以显示紫色，而第1 列单元（5 5 H）是把图形坐标奇数列置成1 ，最高位为0 则显示为绿色，同时可以从表二中发现第6 、7 两列同时为1 ，所以中间有一小段白色水平线。

由此可以看出：偶数列单元的位数奇偶性与图象列数的奇偶性是一致的，而奇数列单元的位数奇偶性与图象列数的奇偶性是相反的，所以，必须把偶数列单元的低七位二进制值右移一位（对紫色和蓝色）或左移一位（对绿色和橙色）以后，变为奇数列单元低七位二进制值，才能实现同一行上单一色彩的一条水平线。如表二中2 0 0 1 H单元置成0 0 1 0 1 0 1 0（2 AH）才能得到一段"连续"的紫色水平线。例如，APPLE SOFT语言中用HCOLOR＝2：HPLOTX1，YTOX2，Y语句，来作单一彩色的水平线时，上述"奇偶性"问题，是由解释程序自动解决的。

由于在图象的偶数列上是无法显示绿色或橙色的，而在奇数列上是无法显示紫色或蓝色的。所以，当企图作一段斜线（或者曲线和圆）时，解释程序是按分段近似方法来处理的，结果可能会把设想的连续白色斜线分成紫色和绿色（HCOLOR＝3 时）线段或者是蓝色和橙色（HCOLOR＝7 时）线段显示出来，而把设想的蓝色的圆，显示成断断续续的弧段，因为，圆上的一些象素可能落在奇数列上，而无法显示。

三．彩色造型

掌握了上述规律后，就可以根据要求的不同坐标、大小、色彩来设计各种图形了，包括各国各民族的文字、数字、各种符号标志等。只要把图形的点阵描绘出来，找出它在图形区对应的列单元地址，以及对应位数，并考虑其色彩要求，置成对应的数值，就可以达到预期的目的。称之为彩色造型。

列单元 / 列数 / 行数	第0列单元							第一列单元								
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
第0行																
⋮																
第7行										(1)						
第8行								(2)		(3)		(4)				
第9行						(5)		(6)		(7)		(8)		(9)		
第10行										(10)						
第11行										(11)						
第12行										(12)						
第13行										(13)						
第14行																

图二　一个蓝色箭头的点阵

例如要在第一页图形区、以第7 行第9 列为顶点作一个橙色箭头，首先按行列描出该标志的点阵，如图二。

要注意，橙色只能在奇数列上有点阵（共用1 3 个点组成），而且有关的列单元最高位均为1 。接着计算这些点阵所在列的单元地址及对应的内容（数值），将结果列于表三：

表三

象素号	列单元地址	二进制值	十大进制值
1	3C01H	10000100	84H
2. 3. 4	2081H	10010101	95H
5	2480H	10100000	A0H
6. 7. 8. 9	2481H	11010101	D5H
10	2881H	10000100	84H
11	2C81H	10000100	84H
12	3081H	10000100	84H
13	3481H	10000100	84H

要特别注意图形可能是跨列、跨段、跨区的，必须符合上述内存分配规律。同时彩色要受"奇偶性"规律的限制，例如上例，若要求作蓝色箭头标志，那末顶点就不能设置在奇数列，必须稍作移动。

然后，利用APPLE SOFT语言中POKE语句或在监控状态下，直接向内存的列单元置数，均可以得到理想的彩色造型效果，当然屏幕先要设置成对应高分辨作图格式。

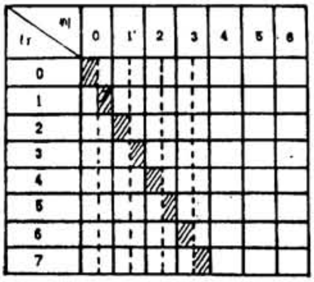
四．把水平方向分辨率提高一倍

仔细观察会发现每个列单元的最高位是控制其低七位的色相的，即控制光点在一个象素列的左半列（最高位为0 时）或右半列（最高位为1 时），这在彩色显示器上可以使一列上有不同的两种色彩，但在黑白显示器上则可以使一列上有两个处于左右半列的光点，从而可使水平方向的分辨率由2 8 0 个象素提高到5 6 0 个象素。

例如，从第0 行0 列到第7 行3 列作一条斜线，把对应列单元地址内存中置以相应的数值。注意，每列单元在一行上仅有一个光点，每一象素列占两行，但用最高位来控制它在该列的左半列或右半列。如图三所示。

这样使这一经过4 列的斜线由8 个光点组成，使图形分辨率提高了一倍。当然对于每一列单元仅有一个光点，实现这样的控制较为容易。如果每一列单元中有多个光点，那末其最高位对该列所有低七位是同时控制的，不可能使一个列单元中某些位对应象素在左半列而另一些位对应的象素在右半列。这就限制了这一方法的使用范围。

列单元地址	十大进制值
2000H	01
2400H	81
2800H	02
2C00H	82
3000H	04
3400H	84
3800H	08
3C00H	88



图三　一条斜线

五．结束语

了解并掌握高分辨率造型和色彩的这些规律以后，可以用APPLE SOFT语言或机器语言，在内存的两个高分辨率图形区内编制相应的各种造型软件，也可在磁盘上建立图形库供调用，甚至可以在APPLE－II系统中配置双显示器：一个用来显示图形；另一个用来显示程序和数据，还可以配置彩色绘图仪等外设，构成一个完整的彩色CAD系统。

敬告读者

本刊因人手关系，决定从下半年起不办理８７下半年杂志邮购业务。本刊１９８６年特刊仍存少许，需购者请参阅本刊今年第二期第３页"邮购消息"，于上半年汇款给本社编辑部收。

今年下半年报刊的征订工作已经开始，敬请只订阅上半年本刊的读者速到当地邮局办理订阅手续，本刊代号４６—115。

一九八六年广东省青少年计算机程序设计竞赛部分试题的评述及参考解答

——华南师范大学计算机科学系　李冠英　邹家炜——

一、小学组笔试部分的第二题。

题目：

判别以下各题是否正确，正确划“√”，错误的划“×”，若是错误的请改正。（不划者不计分）

其中的第4 小题：

（4）　10　PRINT　A=5　（　　）

这个语句不少人认为是错误的，其理由是不能打印赋值语句或者说不能这样打印赋值语句。而事实上只要我们上机试一试，分析一下便可知道这个程序行是没有错误的。

```
10  PRINT  A=5
RUN  回车
0
```

这里的A＝5 不应该看成是赋值语句，而应该理解为是关系表达式，显然这个语句的功能是打印A＝5 这个关系表达式的值，由于预先没有对A赋值故A的当前值为0 ，这样A＝5 这个关系是假的。故A＝5 这个关系表达式的值为0 。也许有人问，这个问题不是具有二义性吗？不。首先我们来看打印语句的通用格式：

[行号] PRINT ["字符串"] ； [变量或表达式] [；或，]

第一项　第二项　第三项　第四项

个数可以选择

而最基本的格式：

[行号]　PRINT　[变量或表达式]

可见语句1 0 是符合这个格式，而且我们并不是孤立地看A＝5 ，而是严格地运用BASIC语言中的语法和词法的格式和规则从整个语句的语意来看待A＝5 的。这样显然不会存在二义性。正如

20　IF　B=5　THEN　C=5　语句，我们不会说这里的B＝5 是赋值语句，也不会说C＝5 是关系表达式。

二、中学组复试笔试部分的第三题。

题目：

3．编写能解决一笔画问题的通用BASIC程序，对程序的要求：

①具有通用性

②能解决你自定的一个图形

③这个图形应具有6 个或6 个以上的节点。

注：这个图形要在卷上画出来。

这道题从卷面看，有些同学错误地写出了在高解析度或低解析度的状态下作出一个能够一笔画的图形的程序来，这是不符合题意的，这部分同学主要不懂得图论中的一笔画问题。但也有些初中的同学能写出一笔画问题的判定定理，他们的答卷是有思路的，可惜往往在进行画时出错，例如对出发点的选取，如何保证从一条边进入，从另一条边出去。在程序中的数据如何决定……等，在此我们想详细地分析这个题目。

1．什么叫一笔画问题呢？

这个问题是图论中最古老的问题之一。

könlsbeg 是东普鲁士的一座城，第二次世界大战后划归苏联所有，也就是现在的加里宁格勒，pregel 河流经过这个城市，如图1 所示，A，B是Pregel 河的两岸，C、D是河中的两个弧岛，C、D两岛与两岸，以及彼此之间由七座桥相连，有人提出过这样一个问题：从一个地方出发，通过每座桥一次，而且仅仅通过一次，最后回到原来的地方，问这样的路经是否存在？

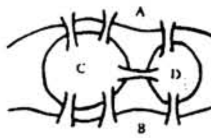


图1

这个问题的提出虽然是出自于游戏，但它的数学模型有着实际意义，由于Euler 解决了这个问题，故此也称为Euler 回路问题。于是这个七桥问题就变成了通常所说的一笔画的问题。

简单地说由若干个不同的顶点与连结其中某些顶点的边所组成的图形，能否一笔画出这个图（每条边都无遗漏也无重复地画到）？或能否一笔画出这个图并且最后可以回到原来的出发点？

2．如何判别一个图形能否一笔画？

在图论中有这样一个定理：

有限图G是一条链（即可以一笔画成）的充分必要条件是G是连通的，且奇顶点的个数等于0 或2 。并且当且仅当奇顶点的个数为0 时，连通图G是一个圈。（弧立点可以看做圈）有兴趣的读者可

以参考上海教育出版社的《趣味的图论问题》一书。为了让同学们能比较直观地了解这个问题，在这里我们把判别的步骤说一说：

1° 我们把图形各边交点叫做顶点。奇数条边的交点叫奇顶点，偶数条边的交点叫偶顶点。

2° 如果该图形的奇顶点超过两个时，则不能一笔画。

要注意的是我们现在所研究的图形是假定为“连”在一起的(即连通的)。其次我们可以证明对于任意的图G，奇顶点的个数一定是偶数。这样七桥问题如图2，有4个奇顶点因而不能一笔画成，也就是说一个旅行者要既无重复也无遗漏地走过图1的七座桥是不可能的。

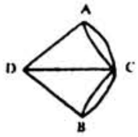


图2

3. 若图形可以一笔画，那如何画呢？

1° 选好出发点：

若奇顶点为0时(设有奇顶点)，则任选一个偶顶点作出发点。

若奇顶点为2时，则选一个奇顶点作出发点。

2° 画的原则是保证从一条边“进入”，从另一条边“离去”。事实上从其中一个奇顶点(假使恰好只有两个奇顶点)出发并追踪这条路，每当我们从一以前未曾走过的边“进入”一顶点时总存在一条以前未曾走过的边以供“离去”。

4. 如何在计算机上实现呢？

下面来谈谈写程序要考虑的几个问题。

如下图3。

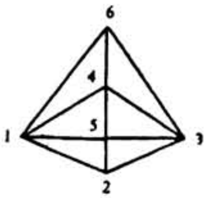


图3

1° 由上图可见其顶点数为6(N=6)对于这N个点，可以进行数据的组织，和编码不妨建立下面的一张表。

	1	2	3	4	5	6
1	0	1	0	1	1	1
2	1	0	1	0	1	0
3	0	1	0	1	1	1
4	1	0	1	0	1	1
5	1	1	1	1	0	0
6	1	0	1	1	0	0

注：考虑某一个顶点向其他顶点的连线是否存在(不重复，不重叠)

若有连线的填上1，无连线则为0，当然该点自身的连线是不存在故也为0。

2° 由图形和上面的表可统计出每个顶点I引出的边的条数，称为I的次数，记作deg I 故如图

deg 1=4，deg 2=3，deg 3=4，deg 4=4，deg 5=4，deg 6=3。

3° 判别每一个顶点的次数，并求出奇顶点的总个数，若奇顶点的总个数大于2则打印出“无解”而结束，即不能一笔画。若奇顶点的总个数小于等于2，则进行一画笔。

4° 如选定第6个顶点(奇顶点)作为出发点向其他各顶点逐一搜索。(若搜索到第I个顶点为零则向第I+1个顶点搜索)。这里第1个顶点就不为零的，这时判别第1个顶点的次数是否大于1，若大于1，则说明应离去第6个顶点向第1个顶点进入(画出第一个笔画)，然后把第1个顶点的次数减1，(第6个顶点的次数也减1)，这时总的次数减2，又把第1个顶点作为出发点，去找下一个顶点为进入点……如此循环。

5° 当第I个顶点的次数不大于1时，判别总的次数是否大于2，若大于2则转判第I+1个顶点作为进入点……。若总的次数不大于2则印出当前的顶点(画出最后一笔)而结束。

设：N____顶点个数。

R(N，N)____每个顶点编码。

D(N)____每个顶点的次数。

H____统计奇顶点的个数。

B____一笔画的起始顶点，(又作当前顶点)。

T____各个顶点次数的总和。

对于图3可编出下面的程序：

```
10 INPUT "N="; N
20 DIM R(N, N), D(N)
30 B=1
```

```
   40 FOR I=1 TO N
   50 FOR J=1 TO N
   60 READ R(I,J)
   70 PRINT R(I,J);",";
   80 D(I)=D(I)+R(I,J)
   90 NEXT J
   100 PRINT
   110 T=T+D(I)
   120 IF D(I)/2=INT(D(I)/2)
THEN150
   130 H=H+1
   140 B=I
   150 NEXT I
   160 IF H>2 THEN PRINT "NO
 SOLUSION":END
   170 PRINT B;
   180 I=0
   190 I=I+1
   200 IF R(B,I)=0 THEN 190
   210 IF D(I)>1 THEN 250
   220 IF T>2 THEN 190
   230 PRINT "-";I;
   240 END
   250 PRINT "-";I;
   260 R(B,I)=0 : R(I,B)=0
   270 D(I)=D(I)-1  D(B)=D(
B)-1
   280 T=T-2
   290 B=I
   300 GOTO 180
   310 DATA 0,1,0,1,1,1,1,0,
1,0,1,0,0,1,0,1,1,1,1,0,1,0,
1,1,1,1,1,1,0,0,1,0,1,1,0,0
RUN [回车]
N=6 [回车]
0,1,0,1,1,1,
1,0,1,0,1,0,
0,1,0,1,1,1,
1,0,1,0,1,1,
1,1,1,1,0,0,
1,0,1,1,0,0,
6-1-2-3-4-1-5-3-6-4-5-2
```

三．中学组复试上机操作部分的第二题

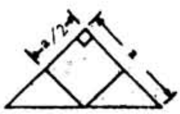
题目：一等腰直角三角形，腰长为a（如下图）从三角形中依次分离出面积为$(a/2)^2$，$(a/4)^2$，$(a/8)^2$…的正方形直到剩余的面积等于或小于原面积的10%为止，问至少可以分离出多少个正方形？有人编出以下程序，请你上机调试好。

要求：1．将程序全部输入计算机。

2．改正程序中的错误。

3．并回答至少可以分离出多少个正方形。

```
10 INPUT A
20 S0=0.5*A^2
30 I=1,K=0
40 A=A/2
50 S0=S0-A^2*I,K=K+1
60 I=I+2
70 IF S0>0.1*S0 THEN 40
80 PRINT "K=";K
90 END
```



从考場上看大部分同学的上机操作的基本技能是比较熟练的。其表现在输入速度快，人机对话，对错误信息的处理迅速，绝大部分的同学能顺利地完成第一题。而第二题是具有一定的难度，因为它的错误之处是比较隐蔽，程序的错误不是出自语法错误，因而机器不会提供错误信息，全靠同学们的敏锐和细心去发现程序中算法的错误，从而通过上机调试把程序中的错误改正。在调试中我们发现有些同学由于没有注意到题目中的“剩余的面积等于或小于原面积的10%为止。”的要求，又疏忽于70语句中的S0>0.1*S0，这是失败的原因之一，其次是对每次分离的做法思路不清，而未能把60语句改为I=I*2，为什么要这样改呢？事实上第一次分离时，是用原三角形的面积减去一个边长为(a/2)的正方形的面积，第二次分离时，是用第一次分离后剩下的二个三角形面积相应地减去二个边长为(a/4)的正方形的面积。第三次分离时，是用第二次分离后剩下的四个三角形面积相应地减去四个边长为(a/8)的正方形的面积，显然I这个参数是依照1，2，4，8……这个规律变化的，这就是我们把60语句的I=I+2改为I=I*2的原因。

参考答案：

1° 应加入 25 S1=S0

2° 第60语句应改为 60 I=I*2

3° 第70语句应改为 70 IF S0>0.1*S1 THEN 40

4° 至少可分离出15个正方形

绘Z＝F（X，Y）立体图的一个简化程序

中国人民解放军5720工厂　李兆丁

Z＝F（X，Y）的立体图在教学中有一定用途，但通常程序复杂，绘制时间长，对硬件的要求较高，这里提供一个用BASIC语言编的简化程序，语句不多，绘制时间比通常的要缩短一倍，在LASER310的PP40描绘器上即可实现，稍加修改还可用于PC－1500与苹果机上，绘出的图形虽不够精确，但尚可用于教学，下面将此程序做一简要说明，着重于如何使用，程序清单附在最后。

X1、X2、Y1、Y2、：Z＝F（x，y）的定义域。

V1、V2、V3、V4：规定图形在打印纸上的位置，见图一。

TH、PH：平行光与物体坐标原点所成的角度（应选择在0～180°之间），TH是绕Z轴旋转的角度，PH是绕X轴旋转的角度，见图一。

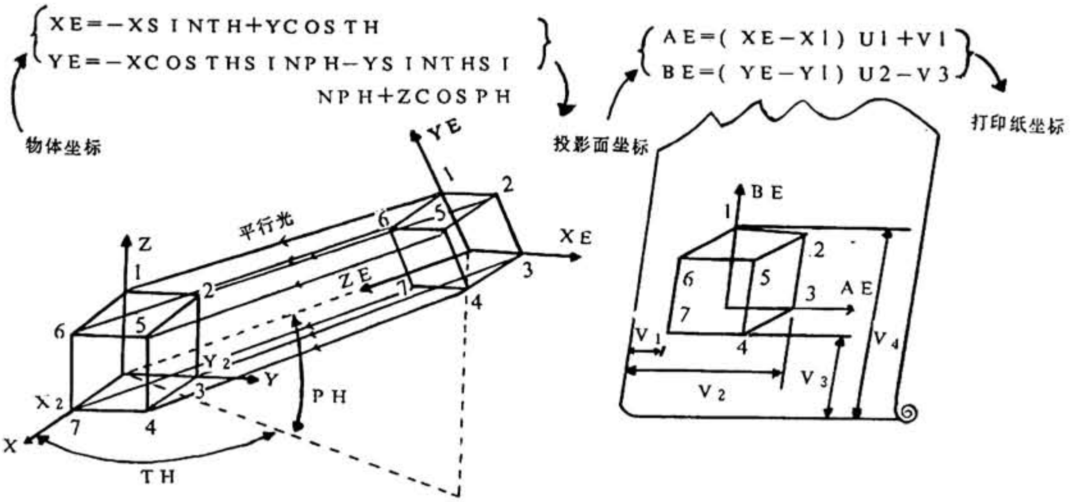
L0：规定图形由几根曲线组成。

P0：规定每根曲线由几个点组成。（每点之间用直线连接）。

XE、YE：投影平面坐标，见图一。

AE、BE：打印纸坐标，见图一。

X3：曲线上每个点之间的间隔（增量）。



图一

Y3：曲面上每根曲线之间的间隔（增量）。

U1：打印纸上的图形，在AE方向与物体在X方向的大小比例。

U2：打印纸上的图形，在BE方向与物体在Y方向的大小比例。

200－210子程序：为PP40描绘器的具体程序，若将此程序用于PC－1500或苹果机则需将这两句连同第10、60、180句稍加修改。

300－320子程序：求Z＝F（X，Y）平行投影于二维平面的坐标XE、YE，以及求XE与YE转换成打印纸坐标AE、BE。

30－40：将角度化为弧度，并为300－320子程序简化记号。

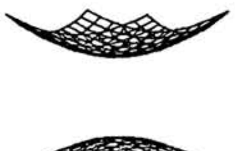
80－120：绘制组成曲面Z＝F（X，Y）的单组曲线。

130：问要不要网状曲线，若需要则输入C＝1，否则输入C＝2。

140－170：绘制组成曲面Z＝F（x，y）的网状曲线。

使用此程序时注意的几个问题：

1，凡属Z＝SQR（……）类型的方程，如球面Z＝SQR（$1-x^2-y^2$），双曲面Z＝SQR（$1+x^2+y^2$）等，因计算机对于开方的结果只取正值，所以在运行完毕后，应将300句改为Z＝－SQR（……），再次运行，图二是双曲面的图形，若不取Z的负值，再次运行，只能画出双曲面的上面一支。



```
20 X1=-.7:X2=.7:Y1=-.7:Y2=.7:P0=11:L0=13
:TH=120:PH=20
60 V1=100:V2=250:V3=0:V4=150
200 LPRINT"M";INT(A1);",";INT(B1*.05)
205 LPRINT"D";INT(A2);",";INT(B2*.05)
300 Z=20*SQR(1+X*X+Y*Y)
  FOR THE LOWER PART : 300 Z=-20*SQR(1+X
*X+Y*Y)
```

图二

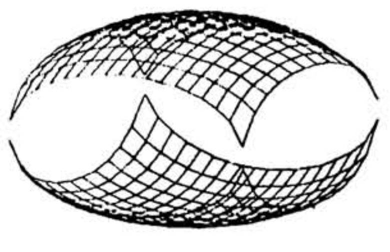
2，若U1＝U2，物体与输出的图形相似，无变形，否则将产生变形，图三的曲面方程为 Z＝SQR（$1-x^2-y^2$），U1＝U2是一球面。



```
20 X1=-.7:X2=.7:Y1=-.7:Y2=.7:P0=13:L0=11
:TH=120:PH=20
60 V1=250:V2=400:V3=100:V4=250
300 Z=SQR(1-X*X-Y*Y)
  FOR THE LOWER PART : 300 Z=-SQR(1-X*X-
Y*Y)
```

图三

绘制图四的程序与图三的相同，仅V1有改变，这时U1≠U2了，产生了变形，成了椭球面。

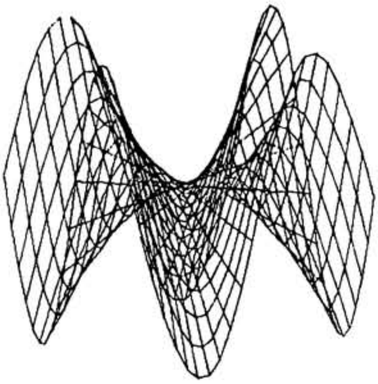


```
20 X1=-.7:X2=.7:Y1=-.7:Y2=.7:P0=17:L0=11
:TH=120:PH=20
60 V1=150:V2=400:V3=100:V4=250
300 Z=SQR(1-X*X-Y*Y)
  FOR THE LOWER PART : 300 Z=-SQR(1-X*X-
Y*Y)
```

图四

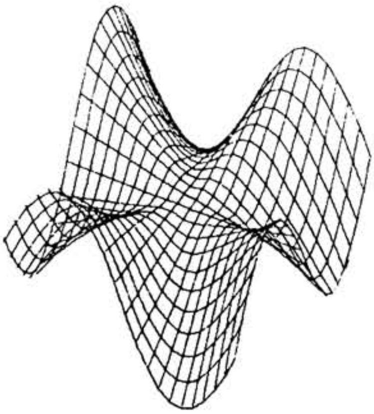
3，适当指定P0与L0的值可避开函数的奇点，例如Z＝xy（x^2-y^2）/（x^2+y^2），当x＝0，y＝0时，Z→∞，可见原点是函数的奇点，若取P0＝20当x从－4到4，增量为x3＝（x2－x1）/P0＝0.4时，逐点扫描到第十点x＝－4＋0.4×10＝0，正好落在x＝0上，y同样如此，若取P0＝21则可跨越奇点，为了避开奇点，通常令P0与L0为奇数。

4，当图形在某一TH与PH的角度下看不清时，可另选角度，例如图五是Z＝xy（x^2-y^2）/（x^2+y^2）的图形，在TH＝60，PH＝15下，看不清，可另取TH＝PH＝120，见图六。



```
20 X1=-4:X2=4:Y1=-4:Y2=4:P0=21:L0=21:TH=
60:PH=15
60 V1=150:V2=400:V3=150:V4=400
300 Z=X*Y*(X*X-Y*Y)/(X*X+Y*Y)
```

图五

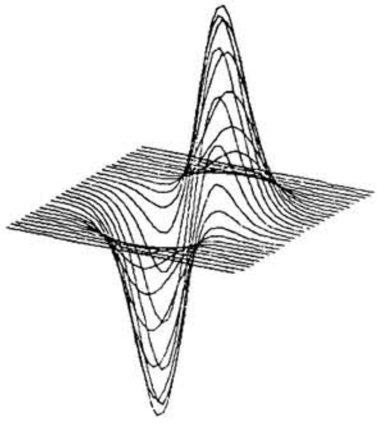


```
20 X1=-4:X2=4:Y1=-4:Y2=4:P0=21:L0=21:TH=
120:PH=120
60 V1=150:V2=400:V3=150:V4=400
300 Z=X*Y*(X*X-Y*Y)/(X*X+Y*Y)
```

图六

TH=45，PH=35·2644得到正等轴侧图，TH=22·2077，PH=20·7048 得到正二等轴侧图。

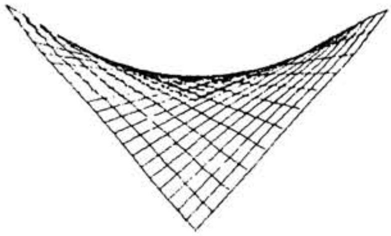
5，并非所有的图形都要画成网状曲线，以增强立体感，某些图形如图七用单组曲线表示时已能看清，则令C＝2 即可。



```
        C=1OR2? 2
20 X1=-3:X2=3:Y1=-3:Y2=3:P0=41:L0=22:TH=
120:PH=20
60 V1=150:V2=400:V3=150:V4=400
300 Z=-8*EXP(-X*X-Y*Y)*(X+Y)
```

图七

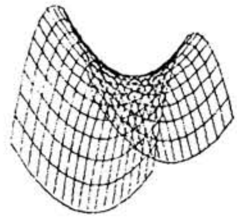
6，计算机可自动进入直纹面，见图八的双曲面抛物面Z＝x y 。



```
20 X1=-3:X2=3:Y1=-3:Y2=3:P0=11:L0=21:TH=
45:PH=120
60 V1=150:V2=400:V3=150:V4=250
300 Z=X*Y
```

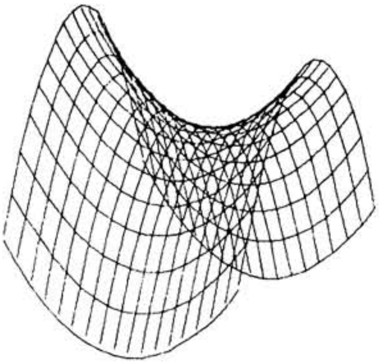
图八

7，打印纸上输出图形的大小可有三种方法调节。(1)改变打印纸上V1、V2、V3、V4的数值，例如对双曲抛物面$Z=x^2/4-y^2/2$，在V1＝50，V2＝200，V3＝0，V4＝150下，图形高40 mm，宽42 mm，见图九；在V1＝50，V2＝300，V3＝0，V4＝250下，图形高65 mm，宽70 mm，见图十。



```
20 X1=-3:X2=3:Y1=-3:Y2=3:P0=17:L0=17:TH=
120:PH=20
60 V1=50:V2=200:V3=0:V4=150
300 Z=X*X/4-Y*Y/2
```

图九



```
20 X1=-3:X2=3:Y1=-3:Y2=3:P0=17:L0=17:TH=
120:PH=20
60 V1=50:V2=300:V3=0:V4=250
300 Z=X*X/4-Y*Y/2
```

图十

⑵若需将图形放大（缩小）的倍数较大时，常用直接法，这时将2 0 0 句改为

LPRINT "M"; INT(A1 /2); ","; INT(B1 /10)

将2 0 5 句改为 LPRINT "D"; INT(A2 /2); ","; INT(B2 /10) 输出图形宽度将缩小半倍，高度将缩小1 0 倍。例如 $Z=x^3-3XY^2$ 需将高度缩小2 0 倍才行，见图十一。

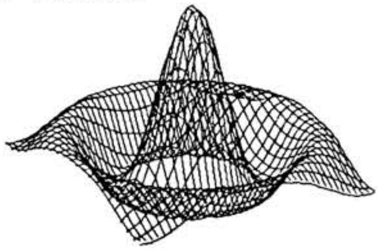


```
20 X1=-4:X2=4:Y1=-4:Y2=4:P0=17:L0=17:TH=
35:PH=120
60 V1=100:V2=350:V3=100:V4=250
200 LPRINT"M";INT(A1);",";INT(B1*.05)
205 LPRINT"D";INT(A2);",";INT(B2*.05)
300 Z=X*X*X-3*X*Y*Y
```

图十一

通常，方程的次数高，需缩小的倍数越大，有一些方程的次数虽不高，但方程的前面有一较大系数，如 $Z=20(1+x^2+y^2)^{\frac{1}{2}}$ 则缩小的倍数也要大一些，见前面图二。

⑶改变物体的定义域也可影响输出图形的大小，但通常不用此法，因为定义域用于确定物体的绘制范围，正如拍照时改变了取景范围往往将某些精彩部分丢掉了，图十二的曲面方程 $Z=10\,SIN\sqrt{x^2+y^2}/\sqrt{x^2+y^2}$ 与图十三的相同，但x 1 =-8，x 2 =8，y 1 =-8，y 2 =8，定义域缩小了，不能反映物体的全貌，不过输出的图形却比图十三大了一些。定义域缩小，其余不变，图形增大，反之亦然。



```
20 X1=-8:X2=8:Y1=-8:Y2=8:P0=39:L0=21:TH=
60:PH=25
60 V1=100:V2=350:V3=100:V4=350
300 Z=10*SIN(SQR(X*X+Y*Y))/SQR(X*X+Y*Y)
```

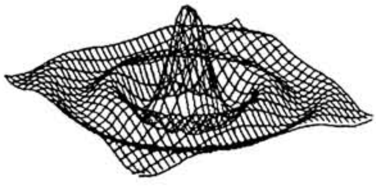
图十二

程序清单：

```
10 CLEAR:LPRINTCHR$(18),"C2"
20 X1=-13:X2=13:Y1=-13:Y2=13:P0=39:L0=21
:TH=60:PH=25
30 K0=3.14159/180:TH=TH*K0:PH=PH*K0:S1=?
IN(TH):C1=COS(TH)
40 S2=SIN(PH):C2=COS(PH):S3=C1*S2:S4=S1*
S2
50 X3=(X2-X1)/P0:Y3=(Y2-Y1)/L0
60 V1=100:V2=350:V3=100:V4=350
70 U1=(V2-V1)/(X2-X1):U2=(V4-V3)/(Y2-Y.?

80 FORU=0TOL0:Y=Y2-U*Y3:X=X1:GOSUB300
90 A1=AE:B1=BE
100 FORV=0TOP0:X=X1+V*X3:GOSUB300
110 A2=AE:B2=BE
120 GOSUB200:NEXT:NEXT
130 INPUT"C=1OR2";C:IFC=1THEN140
135 IFC=2THEN180
140 FORU=0TOP0:X=X1+U*X3:Y=Y2:GOSUB300
150 A1=AE:B1=BE
160 FORV=0TOL0:Y=Y2-V*Y3:GOSUB300
170 A2=AE:B2=BE:GOSUB200:NEXT:NEXT
180 LPRINT"A":END
200 LPRINT"M";INT(A1);",";INT(B1)
205 LPRINT"D";INT(A2);",";INT(B2)
210 A1=A2:B1=B2:RETURN
300 Z=10*SIN(SQR(X*X+Y*Y))/SQR(X*X+Y*Y)
310 XE=-X*S1+Y*C1:YE=-X*S3-Y*S4+Z*C2
320 AE=(XE-X1)*U1+V1:BE=(YE-Y1)*U2-V3:RE?
TURN

RUN
```



C=1 OR 2? 1

图十三

为充分利用打印纸而设计的程序

广州二中　简志雄　傅雄俭

APPLE Ⅱ的监控程序，为我们提供了不少命令，其中有检查存储器内容和反汇编列清单两项，均是用以输出的。若在显示器中显示，则效果十分令人满意。但不幸的是，若用打印机输出，那么就有明显的弊端：打印的内容均落在打印纸的左边，右边空开一大段，十分不美观。而且，这种方式使打印纸被大大地浪费，这是尤其使人不能容忍的。

为了充分利用打印纸，达到美观的效果，同时便于查阅，便于装订，我们编了下面两个程序，它们的特点是简单，实用，效果也很好。下面分别介绍其功能和用法：

1．PROGRAM1 是用以检查存储器内容的。它由一段BASIC语言的主程序和一段放在＄300后的机器语言子程序组成。运行开始，计算机要求输入要检查的存储单元的首、末地址，你必须输入两个四位的十六进制数。计算机接着就会用打印机输出你需要的资料，并且是以每十六个为一行，表头还打上0～F的数字，既美观又方便检查。程序列印中已附了一个执行结果，读者可从中见到其执行后输出的＄6000～＄60DA的一段存储单元的内容，确是十分美观的。必须声明一点的是：类似的程序，以前已有人编过，而笔者所见的，均不及这个程序来得简短，快捷。如，香港的COMPUTING AGE就曾发表过一个类似的程序，但其长度为本程序的三倍，而运行速度却十分慢。

2．PROGRAM2 是用以反汇编列清单的，使用时同样要输入首、末地址（四位的十六进制数）。程序运行时会把汇编程序分成两列列出，若每列长度超过四十行，还会自动换页，结束时若不足四十行，则打足空行成页。页间还会打出横线，便于裁剪成页，装订成册。本程序同样以一个BASIC主程和一个放在＄300后的机器语言子程组成。运行的结果是把本程序的机器语言子程列了出来，读者可从中看出执行的效果。

电脑运动鞋面世

香港　何竞欧

近来，国外兴起了一种与电脑结合的运动鞋，使球鞋、跑步鞋、网球鞋、跳高、跳远等运动鞋实现电脑化。这是运动员的一种新装备，也是体育用品研究家的最新杰作，体育用品从此进入了一个新纪元。

一家美国体育用品公司，为了研制电脑运动鞋，拨出了一亿七千万美元作为发展经费，已获得初步成功。例如，ADI DAS运动鞋厂最近推出的一种电脑跑步鞋，它将电脑计算器藏在鞋跟里，可以把穿着者每一步的压力、步幅、时间和热量等资料记录在内，然后将该软件抽出，放入电脑中，便可计算出运动员消耗了多少卡路里热量，速度、步幅如何，步重多少，跑了多少时间和距离，等等；还可以同以前储存的资料对比，分析出是否进步或退步。这些资料，又可供体育研究家作研究分析用，指示运动员应如何改善运动的质与量。此外，还可以根据各人的健康状况，事先输入运动量界限，一旦运动量达到界限，鞋跟的仪器会发出警告的响声，这对于一些心脏或其他身体有问题的人来说特别适用，不致于因运动过量而出事。

电脑跑步鞋使用的电脑，体积非常细小，重量极微，运动员难以感觉得到。有的把电脑安装在运动鞋尾部，而ADI DAS厂所生产的电脑运动鞋却把它安装在鞋面上，如图所示。由鞋面上电脑所控制的传感器十分灵敏，可以及时记录得到距离、时间、速度、压力、热量等参数。这种小巧电脑，贴在鞋面，上有软垫，不会梗脚，脚部也绝无受压之弊，对跑步当然更无影响。目前，这种电脑跑步鞋每双约售600港元。随着各种电脑运动鞋的推广应用，售价会逐渐降低，其他的体育用品也将会越来越多地与电脑相结合。



为增强适应性，以上两个程序的机器语言子程均是可改变入口而照样执行的程序，读者可根据实际情况改变入口，再改变第5 行的入口值则可。

以上所述的两个程序都力求简短、实用，并求得最快的运行速度。在程序设计上，有不少巧妙的地方，有兴趣的读者可认真找出其中的妙处来，此处不一一详述了。

程序1 ：

```
5 L   768
10  HOME : VTAB 5: INPUT "START ADDRESS IN HEX.";A$: GOSUB 60: POKE L + 1
    ,C    INT (C / 256) * 256: POKE L + 5, INT (C / 256):C$ = A$
20  VTAB 9: INPUT "  END ADDRESS IN HEX.";A$
25  GOSUB 60: POKE L + 9,C    INT (C / 256) * 256: POKE L + 13, INT (C /
    256): PR# 1: PRINT : PRINT  TAB( 7);"0  1  2  3  4  5  6  7  8  9  A
     B  C  D  E  F  "
30 B$ =  RIGHT$ (C$,1)
35  IF B$ = "0" THEN 50
40  PRINT : PRINT C$;"-";: FOR I = 1 TO  ASC (B$) - 47 - 7 * (B$ >  = "A"
    ): PRINT "   ";: NEXT
50  CALL L: PRINT : PR# 0: END
60 C = 0: FOR I = 1 TO 4
70 B$ =  MID$ (A$,I,1):A =  ASC (B$) - 48 - 7 * (B$    = "A")
75 C = C + A * 16 ^ (4 - I)
80  NEXT : RETURN

]
0300-   A9 41       LDA   #$41
0302-   85 3C       STA   $3C
0304-   A9 03       LDA   #$03
0306-   85 3D       STA   $3D
0308-   A9 51       LDA   #$51
030A-   85 3E       STA   $3E
030C-   A9 03       LDA   #$03
030E-   85 3F       STA   $3F
0310-   A0 00       LDY   #$00
0312-   A5 3C       LDA   $3C
0314-   29 0F       AND   #$0F
0316-   D0 03       BNE   $031B
0318-   20 92 FD    JSR   $FD92
031B-   A9 A0       LDA   #$A0
031D-   20 ED FD    JSR   $FDED
0320-   B1 3C       LDA   ($3C),Y
0322-   20 DA FD    JSR   $FDDA
0325-   20 BA FC    JSR   $FCBA
0328-   90 E8       BCC   $0312
032A-   60          RTS

*
]RUN
START ADDRESS IN HEX.6000
  END ADDRESS IN HEX.60DA

       0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F

6000- A9 08 85 FC A9 0C 85 FD A9 14 85 FE A9 12 85 FF
6010- A9 40 85 E6 A9 53 A0 00 A2 85 20 11 F4 20 B5 60
6020- A6 FC 20 D5 F4 20 B5 60 CA D0 F7 A5 FC 18 69 08
6030- 85 FC E6 E5 A4 E5 C4 FE F0 0A 90 08 C6 E5 88 E6
6040- FE 4C 68 60 20 B5 60 A5 26 48 A5 27 48 20 04 F5
6050- 20 B5 60 20 04 F5 20 B5 60 20 04 F5 20 B5 60 68
6060- 85 27 68 85 26 4C 32 60 A6 FD 20 04 F5 20 CC 60
6070- 20 B5 60 CA D0 F4 A5 FD 18 69 08 85 FD C6 E5 A4
6080- E5 30 06 C4 FF F0 0A B0 08 E6 E5 C8 C6 FF 4C 20
6090- 60 20 B5 60 A5 26 48 A5 27 48 20 D5 F4 20 B5 60
60A0- 20 D5 F4 20 B5 60 20 D5 F4 20 B5 60 68 85 27 68
60B0- 85 26 4C 7D 60 A5 E5 B1 26 48 38 A5 27 E9 20 85
60C0- 27 68 91 26 A5 27 18 69 20 85 27 60 A5 26 C9 50
60D0- D0 08 A5 27 C9 42 D0 02 68 68 60
```

程序2 :

```
5  HOME :P = 768: POKE 255,0
10  INPUT "START ADDRESS:";A$: GOSUB 110: POKE 6,L: POKE 7,H%
20  INPUT "  END ADDRESS:";A$: GOSUB 110: POKE 8,L: POKE 9,H%
30  CALL P:S = 80 -  PEEK (10):S% = S / 2:S% = S - S%
50  POKE P + 55,S%: CALL P + 46
54  PR# 1: GOSUB 150
60  POKE 10,S:: PRINT : PRINT : PRINT
70  CALL P + 71
80  IF  PEEK (255) = 255 THEN  FOR I = 1 TO 40 - S%: PRINT : NEXT : GOSUB
    150: PR# 0: END
90  PRINT : PR# 0: POKE 6, PEEK (206): POKE 7, PEEK (207): GOTO 30
110 C = 0: FOR I = 1 TO 4:B$ =  MID$ (A$,I,1):A =  ASC (B$) - 48 - 7 * (B
    $ >  = "A"):C = C + A * 16 ^ (4 - I): NEXT :L = C -  INT (C / 256) *
    256:H% = C / 256: RETURN
150  PRINT : PRINT : PRINT : FOR I = 1 TO 80: PRINT "-";: NEXT : PRINT : RETUI

]

--------------------------------------------------------------------------------
```

```
0300-   A9 50      LDA   #$50
0302-   85 0A      STA   $0A
0304-   A5 06      LDA   $06
0306-   85 3A      STA   $3A
0308-   A5 07      LDA   $07
030A-   85 3B      STA   $3B
030C-   A9 01      LDA   #$01
030E-   20 63 FE   JSR   $FE63
0311-   A5 3A      LDA   $3A
0313-   C5 08      CMP   $08
0315-   A5 3B      LDA   $3B
0317-   E5 09      SBC   $09
0319-   10 06      BPL   $0321
031B-   C6 0A      DEC   $0A
031D-   D0 ED      BNE   $030C
031F-   F0 06      BEQ   $0327
0321-   C6 0A      DEC   $0A
0323-   A9 FF      LDA   #$FF
0325-   85 FF      STA   $FF
0327-   A5 3A      LDA   $3A
0329-   85 CE      STA   $CE
032B-   84 CF      STY   $CF
032D-   60         RTS
032E-   A5 06      LDA   $06
0330-   85 3A      STA   $3A
0332-   A5 07      LDA   $07
0334-   85 3B      STA   $3B
0336-   A9 27      LDA   #$27
0338-   20 63 FE   JSR   $FE63
033B-   A0 01      LDY   #$01
033D-   A5 3B      LDA   $3B
033F-   85 0C      STA   $0C
0341-   88         DEY
0342-   A5 3A      LDA   $3A
0344-   85 0B      STA   $0B
0346-   60         RTS
0347-   A5 06      LDA   $06
034D-   85 3B      STA   $3B
034F-   20 D0 F8   JSR   $F8D0
0352-   C6 0A      DEC   $0A
0354-   F0 3D      BEQ   $0393
0356-   20 53 F9   JSR   $F953
0359-   85 3A      STA   $3A
035B-   48         PHA
035C-   84 3B      STY   $3B
035E-   98         TYA
035F-   48         PHA
0360-   A9 27      LDA   #$27
0362-   85 24      STA   $24
0364-   A6 0B      LDX   $0B
0366-   A4 0C      LDY   $0C
0368-   20 99 FD   JSR   $FD99
036B-   A5 0B      LDA   $0B
036D-   85 3A      STA   $3A
036F-   A5 0C      LDA   $0C
0371-   85 3B      STA   $3B
0373-   20 89 F8   JSR   $F889
0376-   20 D3 F8   JSR   $F8D3
0379-   C6 0A      DEC   $0A
037B-   F0 16      BEQ   $0393
037D-   E6 2F      INC   $2F
037F-   18         CLC
0380-   A5 2F      LDA   $2F
0382-   65 0B      ADC   $0B
0384-   85 0B      STA   $0B
0386-   90 02      BCC   $038A
0388-   E6 0C      INC   $0C
038A-   68         PLA
038B-   85 3B      STA   $3B
038D-   68         PLA
038E-   85 3A      STA   $3A
0390-   38         SEC
0391-   B0 BC      BCS   $034F
0393-   68         PLA
```

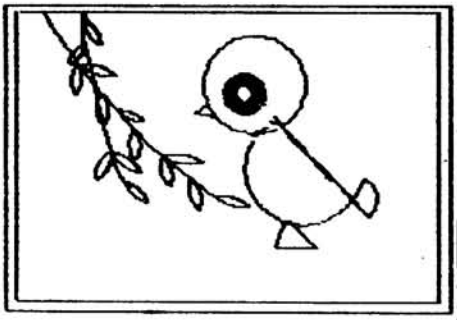
抗干扰稳压器在广州问世

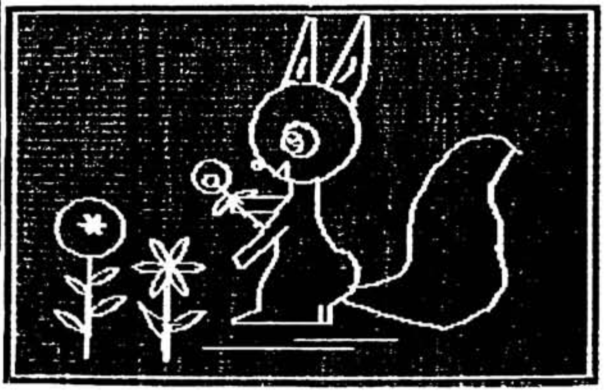
本刊讯　大型计算机的“保镖”——三相抗干扰稳压器最近在中国科学院广州分院电子研究所研制成功，2月20日通过了技术鉴定。

经鉴定测试小组测试和用户提供的报告，该设备除谐波失真外均达到原定设计指标，该设备特点是抑制高能脉冲的能力强，总谐波失真低、音频噪声小、效率高、响应时间短、过载能力强、稳压范围宽。各种指标均达到或超过国外同类产品，该设备是保证计算机等精密仪器安全运转的必备三相供电系统，具有很大的经济价值。

据该机的合作生产单位——广州国营白云无线电厂透露，以单相稳压器的成本核算为基础，预计目前一台三相10KW抗干扰稳压器的生产样机的售价，仅为进口价的50%左右，投入大批生产后，每年为国家节约大量外汇。

电脑艺术





小鸟与松鼠

北京市北京大学14斋　陈旭林

（上接第3页）

与其它科研单位协作研制了火电监控系统；

为电力学院研制了火力发电厂的全仿真系统。

3．它在企业管理方面：

已为万人企业研制了大型企业计算机（网）管理系统；

已为千人企业研制了中型企业计算机（网）管理系统；

已为小型企业研制了计算机管理系统。

4．它在政府办公及所属部门的管理方面：

已为市政府研制了办公自动化的计算机系统；

已为市属财局，市属供电局，市属人事局等研制了计算机管理系统。

5．它在国防建设方面：

已研制了数个重大的计算机工程，有的将接受国家级鉴定，既节约了外汇，又加强了国防建设。

6．它在计算机系统软件方面：

已在小型机及微机的操作系统和语言的移植，改造及研制方面作出很多成绩，并通过了鉴定和验收。

一个百人的软件队伍，在短短三年中，竟作出如此巨大的成就，并为许多同志敬服，实在是值得我们的改革政策研究单位，领导机关，科研部门认真调查研究。

据说它一直负重走在泥泞的道路上，多么希望真正支持改革的领导机关能为他们排除前进道路上障碍，使其在改革道路上迈出更大的步伐!